NBS TECHNICAL NOTE 936

U.S. DEPARTMENT OF COMMERCE / National Bureau of Standards

A FORTRAN Program for Desmearing Small-Angle X-Ray Scattering Curves

QC
100
U5753
No.936
1977
c.2

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau consists of the Institute for Basic Standards, the Institute for Materials Research, the Institute for Applied Technology, the Institute for Computer Sciences and Technology, the Office for Information Programs, and the Office of Experimental Technology Incentives Program.

THE INSTITUTE FOR BASIC STANDARDS provides the central basis within the United States of a complete and consistent system of physical measurement; coordinates that system with measurement systems of other nations; and furnishes essential services leading to accurate and uniform physical measurements throughout the Nation's scientific community, industry, and commerce. The Institute consists of the Office of Measurement Services, and the following center and divisions:

Applied Mathematics — Electricity — Mechanics — Heat — Optical Physics — Center for Radiation Research — Laboratory Astrophysics[2] — Cryogenics[2] — Electromagnetics[2] — Time and Frequency[2].

THE INSTITUTE FOR MATERIALS RESEARCH conducts materials research leading to improved methods of measurement, standards, and data on the properties of well-characterized materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government agencies; and develops, produces, and distributes standard reference materials. The Institute consists of the Office of Standard Reference Materials, the Office of Air and Water Measurement, and the following divisions:

Analytical Chemistry — Polymers — Metallurgy — Inorganic Materials — Reactor Radiation — Physical Chemistry.

THE INSTITUTE FOR APPLIED TECHNOLOGY provides technical services developing and promoting the use of available technology; cooperates with public and private organizations in developing technological standards, codes, and test methods; and provides technical advice services, and information to Government agencies and the public. The Institute consists of the following divisions and centers:

Standards Application and Analysis — Electronic Technology — Center for Consumer Product Technology: Product Systems Analysis; Product Engineering — Center for Building Technology: Structures, Materials, and Safety; Building Environment; Technical Evaluation and Application — Center for Fire Research: Fire Science; Fire Safety Engineering.

THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides technical services designed to aid Government agencies in improving cost effectiveness in the conduct of their programs through the selection, acquisition, and effective utilization of automatic data processing equipment; and serves as the principal focus wthin the executive branch for the development of Federal standards for automatic data processing equipment, techniques, and computer languages. The Institute consist of the following divisions:

Computer Services — Systems and Software — Computer Systems Engineering — Information Technology.

THE OFFICE OF EXPERIMENTAL TECHNOLOGY INCENTIVES PROGRAM seeks to affect public policy and process to facilitate technological change in the private sector by examining and experimenting with Government policies and practices in order to identify and remove Government-related barriers and to correct inherent market imperfections that impede the innovation process.

THE OFFICE FOR INFORMATION PROGRAMS promotes optimum dissemination and accessibility of scientific information generated within NBS; promotes the development of the National Standard Reference Data System and a system of information analysis centers dealing with the broader aspects of the National Measurement System; provides appropriate services to ensure that the NBS staff has optimum accessibility to the scientific information of the world. The Office consists of the following organizational units:

Office of Standard Reference Data — Office of Information Activities — Office of Technical Publications — Library — Office of International Standards — Office of International Relations.

---

[1] Headquarters and Laboratories at Gaithersburg, Maryland, unless otherwise noted; mailing address Washington, D.C. 20234.
[2] Located at Boulder, Colorado 80302.

Standards
1977
QC,
53
936
M

A FORTRAN Program for Desmearing Small-Angle X-Ray Scattering Curves

Technical Note, no. 936

Jacob Mazur

Institute for Materials Research
National Bureau of Standards
Washington, D.C. 20234

U.S. DEPARTMENT OF COMMERCE, Juanita M. Kreps, Secretary

Dr. Betsy Ancker-Johnson, Assistant Secretary for Science and Technology
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, Acting Director

Issued April 1977

National Bureau of Standards Technical Note 936
Nat. Bur. Stand. (U.S.),Tech. Note 936, 83 pages (April 1977)
CODEN: NBTNAE

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1977

For sale by the Superintendent of Documents, U.S. Government Printing Office
Washington, D.C. 20402 - Price $1.45

Stock No. 003-003-01752-3

TABLE OF CONTENTS

Abstract

A FORTRAN program is presented that corrects for collimation effects in small angle x-ray scattering measurements. The method employed in calculations is based on numerical solution of an integral equation, which is written as a Volterra equation of the first kind. This equation is reduced to a system of simultaneous equations, which can be solved by back substitution. Several different algorithms are tried, based on different interpolating polynomials for I(x), the true scattering intensity, which is expressed as a function of the scattering angle x. The problem of numerical stability, which is inherent to the computational algorithms employed in the algebraic solution equation is discussed and exemplified with actual computations.

Keywords: Desmearing; FORTRAN program; integral equation; interpolating polynomial; numerical stability; slit correction; small angle scattering.

CHAPTER 1

1. INTRODUCTION

Measurement of small angle scattering of x-rays is an extremely valuable tool for the study of macromolecular systems. In recent years, these measurements have been extensively employed in order to determine the three dimensional structure of triblock copolymers. These coploymers form domains whose size is ideally suited for small angle x-ray scattering studies. As a consequence, the composition of the domains and the changes of their morphology with variation of stress, temperature or pressure can be easily studied. Since triblock copolymers are frequently used in the plastics, rubber and coating industry, their study and characterizations by small angle scattering is of prime technological importance.

In x-ray scattering measurements, narrow slits are often used in order to collimate the incident beam sufficiently to allow scattering measurements close to the incident beam. Compared to a pinhole of diameter approximately equal to the slit width, the slit system increases the available scattering energy. Therefore, the experimental measurements of the angular distribution of the intensity of small angle x-ray scattering, instead of giving the scattering at a single angle, represent the average intensity over a range of angles. Because of this, the measured scattering curve is often very different from the curve obtained with point collimation. Hence, the collimation effect results in "smearing" of the scattered radiation.

When the slit is very narrow, the effect of the width of the slit on the smearing of x-ray scattering can be neglected, and only the effect of the slit length needs to be considered. In some cases, the slit width produces changes in the scattering curves that should not be neglected. Since the distortion caused by the width of the collimating slit is small, the correction for the width effects might be well approximated using the method developed by Taylor and Schmidt [1]. In this method the measured scattering curve is corrected for the width effect, before it is corrected for the length effect. In the present work, the method of reference [1] is employed. However, more than one width slit correction is being applied to the measured scattering curve. This is done in order to extend the method of reference [1] to cases when the width of the collimating slits is altered at more than one scattering angle.

The extent of the angular region over which the collimation system averages or "smears" the scattered radiation can be expressed by a weighting function. This weighting function represents the intensity distribution within the primary beam in the longitudinal direction. In many cases, with the counter tube measurements, the length of the slit with which the intensity distribution is measured also has an influence on the scattered intensity. If the slit length is equal to k, then the corrected weighting function is equal to the convolution of the uncorrected weighting function with a step function which represents the length of the slit. Thus, the corrected weighting function W is related to the uncorrected weighting function w through

$$W(t) = \int_{-\infty}^{\infty} w(u)h(t-u)du$$

$$h(u) = 1 \qquad 0 \leq u < k \tag{1}$$

$$h(u) = 0 \qquad k < u < \infty$$

w is a complicated function of the geometrical parameters of the collimating system. An excellent review on the method to calculate the weighting function from the geometry of the system is presented by Hendricks and Schmidt [2]. There is another way to determine the weighting function with reasonable accuracy without having to compute it from the geometry of the collimating system. In this method, the weighting function is determined by measuring the intensity of the beam in the plane of registration. This is done by taking various intervals equal to the length of the scanning slit at the detector.

The integral equation which relates the experimentally determined intensity $\tilde{I}(x)$, measured at the scattering angle x, to the true scattering intensity I(x) and to the weighting function W(y) is [3]:

$$\tilde{I}(x) = \int_{-\infty}^{\infty} W(y)\ I((x^2 + y^2)^{1/2})dy. \tag{2}$$

There is a comprehensive literature which deals with solving integral equation (2) [4]. Most of the methods reported involve making certain assumptions about the functional form of W(y) and $\tilde{I}(x)$. An exception is the procedure due to Vonk [5].

In reference [6] the author proposed a class of multistep methods which like Vonk's does not require assumptions about the functional form

of $\tilde{I}$ and W. These methods start with eq. (2) rewritten for positive scattering angle x as a linear Volterra integral equation of the first kind, [6]

$$h(u) = \int_u^\infty K(u,x)I(x)dx \quad 0<u\leq x<\infty \tag{3}$$

where

$$h(u) = \int_u^\infty \frac{xI(x)dx}{(x^2 - u^2)^{1/2}}$$

and

$$K(u,x) = 2x\int_0^1 \frac{W((x^2 - u^2)^{1/2})dz}{(1 - z^2)^{1/2}} \tag{3b}$$

The experimentalist provides smeared intensity data $\tilde{I}(\tilde{x}_0)$, $\tilde{I}(\tilde{x}_1)$, . . . $\tilde{I}(\tilde{x}_n)$ where $0<a = \tilde{x}_0 < \tilde{x}_1 \ . \ . \ . <\tilde{x}_n = b < \infty$. Generally, the x_i are equally spaced. A square grid is set up on [a,b] x [a,b] of the u-x plane with mesh size Δx and mesh points $\{(u_i, x_j) = (i\Delta x, j\Delta x), m\leq i,j\leq N\}$. $m\Delta x$ is the lowest scattering angle at which an intensity measurement is taken. Interpolation is used to obtain values of $\tilde{I}(x_0)$ if measured data at angle x_0 are not available. Quadrature rules are then used to discretize eq. (3). Accordingly,

$$h(i\Delta x) \simeq \sum_{j=i}^{N} a_{ij}\ K(i\Delta x,\ J\Delta x)\ \bar{I}\ (j\Delta x) \tag{4}$$

$$i,j = m,m+1, \ . \ . \ . \ N-1.$$

To complete the set of linear equations we proposed in reference [6] to use the equation

$$h'(N\Delta x) \overset{\sim}{=} \frac{1}{\Delta x}[h(N\Delta x)-h((N-1)\Delta x)]=K(N\Delta x,\ N\Delta x)\bar{I}(N\Delta x). \quad (4a)$$

Here $\bar{I}(j\Delta x)$ denotes an approximation to $I(j\Delta x)$. The set of values $\bar{I}(j\Delta x)$, j=m, . . ., N will be treated as components of the solution vector $\underline{I}$. Ways to approximate $I(j\Delta x)$ and choices of quadrature rules are discussed in Chapter 3 while the evaluation of h is discussed in Chapter 4, Section B. For simplicity, we will frequently drop the bar from $\bar{I}(j\Delta x)$ when denoting approximate values.

The step-by-step discretization of eq. (3) leads to a convergent method if in rather loose terms the difference between the approximate value $\bar{I}(i\Delta x)$ and the true value $I(i\Delta x)$ tends to zero as $\Delta x \to 0$ with $x=i\Delta x$ fixed and $x \varepsilon [a,b]$.

Integral equations of the first kind are known to be ill-posed [7]. This means that small perturbations in h(u) or in K(u,x) may have a large effect on the solution $\underline{I}$. Translated into our physical problem, one could suspect that small errors in the experimental measurements of $\tilde{I}(x)$ or in the collimation weighting function W will be greatly enhanced in the unsmearing process. The extreme sensitivity of the solution vector $\underline{I}$ to errors in the experimental data can be greatly reduced by converting the Volterra integral equation of first kind into the equation of the second kind by differentiation. The disadvantage in making such a conversion is that numerical differentiation of experimental data can introduce serious errors. The method for desmearing experimental data presented in the Technical Note should be employed in cases when numerical differentiation of experimental data is to be avoided.

There are several sources for possible numerical errors. The first one is inherent in the quadrature rule employed to discretize the integral equation. The second one is random errors or noise present in experimental data. Actually, these two sources of errors cannot be treated separately. Certain quadrature rules (e.g. trapezoid rule) require that the data be "sufficiently smooth" to insure numerical stability. (A function is defined to be "sufficiently smooth" if it is continuously differentiable up to sufficiently high order with respect to its argument [8].) However, certain quadrature methods will always be unstable. To get an insight into numerical instability, assume that eqs. (4-4a) are solved by back substitution, starting with $I(N\Delta x)$. The method is absolutely unstable if the successive values of $I(x)$ tend to diverge as Δx is made smaller, while keeping $N\Delta x$ constant. In this work several algorithms for discretization of the integral equation will be employed and which are not absolutely unstable. (These algorithms along with a stability criterion will be presented in Chapter 3.) The purpose of presenting several algorithms is to enable the user of the program to compare results based on various algorithms and to determine whether the maxima and minima present in the unsmeared scattering curve are due to random errors or whether they are real.

The program presented in this Note does not provide for the smoothing of the experimental data. This should be the user's responsibility. In particular, the use of the trapezoidal rule has been known to lead to even-odd oscillations in the computed discrete set of values for $I(x)$ [8]. These oscillations can, however, be reduced by smoothing the results. The method for smoothing the computed $I(x)$ employed in this

work is a modification of the method suggested by Linz [9], and is described in Chapter 3. We also employ an algorithm for discretization of the integral equation which is based on forward differences in a two-step quadrature method and designated as the Nÿstrom algorithm. We found that with this algorithm, no smoothing of the results was necessary, since the computed values for I(x) were relatively free of even-odd oscillations.

There is another source of error that is brought about by the termination of the calculations at a certain point. This termination introduces a finite upper limit to the integral equation (3). The perfect collimation theory calls for an infinite upper limit in this integral equation. This error will be more noticeable in the outer part of the scattering curve, close to its terminal. Therefore, for the results to be physically meaningful, it is important to extend the measurements as far as possible beyond the significant range of the scattering angles. We did find that, if this precaution is taken, the termination error will not affect the locations and the relative intensities of the maxima and minima in the desmeared intensity curve.

To summarize, the procedure employed in this work for the data desmearing, based on a class of step-by-step finite difference methods, has certain advantages and disadvantages over the procedures reported in the pertinent literature. The advantages are two-fold: 1) No numerical differentiation of the experimental data, a process which may introduce serious error, is required; 2) No restriction on the functional form for the weighting function is imposed. The weighting function can be presented as a set of data which are predetermined by the experiment

either by intensity measurements in the plane of registration or by calculation from the geometry of the collimating system.

The main disadvantage of the method of solving eq. (3) by finite differences is that, even with sufficiently smooth data, many quadrature methods, commonly employed in solving finite difference equations, will lead to divergent results as $\Delta x \to 0$.

In this work we provide a few tests of the method of solving eq. (3). These tests are of two kinds: 1) Solution checking and; 2) testing the various algorithms against test functions for $\tilde{I}(x)$, which lead to analytically known perfect collimation intensities I(x). The solution checking calculations are incorporated in the program. Two test functions for $\tilde{I}(x)$ are employed. One is a Gaussian function. The other represents a scattering curve for spheres with uniform electron densities. This function has been tabulated by Schmidt [10]. For the first test function, the perfect collimation intensities are also of Gaussian form. The second test function corresponds to the perfect collimation intensity of

$$I(x) = A \left[\frac{\sin(xa) - xa\cos(xa)}{(xa)^3} \right]^2$$

where a is the radius of the scattering sphere and x is proportional to the scattering angle. In both test functions, a constant weight function was assumed. Both test functions lead to satisfactory results.

The theoretical background for the methods employed in conjunction with the numerical solution of the set of simultaneous equations (eqs. 4-4a) and the ways to cope with the problem of numerical stabilities is

reviewed in Chapter 3. In Chapter 4, the actual computations are described, followed by the FORTRAN program (Chapter 5). In Chapter 6, the various algorithms employed in construction of numerically convergent methods and their effect on the solution vector $\underline{I}$ are discussed and shown in Figures 1-4. The experimental data for this investigation were provided by J. J. Weeks of this laboratory. In Figure 5 we show how unsmearing of data lead to a formation of peaks in the true intensity curve. The data for this Figure were supplied by D. McIntyre of Polymer Institute, University of Akron. At the end of Chapter 6, calculations based on the two test functions described above, are shown (Table 5 and Figure 6).

For the person who is interested only in the preparation and use of the program, we present in Chapter 2 the organization of the FORTRAN program, and provide the instructions for preparation of the input data required for the actual computations.

CHAPTER 2

1. PROGRAM ORGANIZATION

A. Available Programs

There are two separate programs. The first program, listed in Chapter 5, should be used when N, the total number of experimental data employed in calculations exceeds 150. In this case, storing matrices would require very large computer storage. To avoid the storage problem, the matrix elements are computed as functions for each pair of indices i and j, with $i \leq j \leq N$. This process is more time-consuming, than with the matrix storage. Therefore, for $N>150$, if computer time is of prime importance, the weighting function should be approximated by a trapezoid or by a constant, if possible. The case where the weighting function is represented by a set of data is, by far, the most time-consuming.

The second program is designed for $N<150$. Since the components of the matrix employed in calculations (see Chapter 4 for details on the matrices employed) are stored in the computer memory at the beginning of the computations, they can be used in conjunction with various algorithms employed in the computation of the solution vector $\underset{\sim}{I}$. Except for this detail, the two programs are identical. This second program is available upon request. It is not listed here.

B. Instructions and Input Data

The program requires that the components of the input vector $\underset{\sim}{I}_o$ be spaced at equal intervals of x. This increment, Δx, is customarily expressed in milliradians. Very often, the experimentalist, at some point, increases the spacings between two consecutive scattering angles.

Suppose that at a certain scattering angle equal to $p\Delta x$, the spacing between scattering angles is increased from Δx to $\ell\Delta x$, where ℓ is a integer. The computer program will interpolate by quadratic forward interpolation the input data, starting with the angle $p\Delta x$ to give $\tilde{I}$ values at angular increment Δx. The last interval, between $x=(N-1)\Delta x$ and $x=N\Delta x$, is interpolated linearly. The new input vector I will contain $N_t=N_o + (N_o-p)(\ell-1)$ equally spaced angles, where N_o is the number of input values fed into the machine. Only one change in data spacing is allowed.

For example, suppose that the measured data for the scattering intensity were initially collected at a constant interval of $\Delta x=0.2$ mrad. Suppose that the experimentalist, starting with a scattering angle equal to 20 mrads, increased the spacing between consecutive measurements from 0.2 mrads to 1 mrad. If the initial number of experimental measurements was 200, N will be equal to $200+(200-100)\cdot(4)=600$, since $p=100$ and $\ell=5$. The largest scattering angle is equal to 120 mrads.

The lowest scattering angle, at which the initial measurement is taken, ordinarily exceeds the spacing Δx between two consecutive scattering angles. Let m be the ratio of the lowest scattering angle to the spacing between two consecutive scattering angles, rounded to the nearest integer. As a consequence of this rounding off, the measurements of scattering angles might be off by a much as $\Delta x/2$. To avoid this error, the lowest scattering angle should be chosen in such a way that it equals an integer number of the spacing between scattering angles. Thus, $m\Delta x$ measures the lowest scattering angle from which the measurements are taken. The array which represents the input vector $\hat{I}$ after

it is read into the computer, is shifted by m locations.

Let there be g_1 values at initial interval Δx_1 and g_2 values at augmented interval Δx_2, and let $\ell=\Delta x_2/\Delta x_1$. The program will fill vector $\underset{\sim}{I}_o$ and generate vector χ so that one interval Δx will apply everywhere. Only one interpolation is allowed. We have:

$$g_1 = p-m+1, \quad g_2=N_o-g_1.$$

The following diagram 1 (Diagram 1 is presented on page I) shows the preparation of the augmented input vector $\underset{\sim}{I}$. In diagram 1, $x_m=m\Delta x$, $x_{m+1}=(m+1)\Delta x$, etc. Starting angle is equal to $m\Delta x$.

Card 1 provides four input data needed to fill vector $\underset{\sim}{I}_o$. These are: 1) N_o, total number of input data $(=g_1+g_2)$; 2) m, defined so that $m\Delta x$ measures the lowest scattering angle from which measurements, taken at equal interval of Δx, are taken; 3) p, defined so that $p\Delta x$ is the starting angle for change in data spacing and; 4) the number of interpolated values required in each Δx_2 interval. This number is equal to $\ell-1$, where $\ell=\Delta x_2/\Delta x_1$.

These four input parameters are read as NN, ISTART, IENTR and NSUB. They are punched on a single card, FORMAT 4I10.

If ISTART=0, $\tilde{I}_1$ is the intensity at the zero scattering angle (might be obtained by extrapolation of data to zero angle).

The experimentalist might increase the width of the slit at certain scattering angles, in order to provide a sufficient scattering intensity at larger angles. In the computer program, if the card which follows the first card is blank, no correction for the slit-width is performed. For each slit-width correction, a single card is read which provides two sets of data: the lower limit of the scattering angle (in Δx units),

from which the slit-width correction applies, and the value of M_2, the second moment of the slit-width weighting function W_w, defined as:

$$M_2 = \int_{-\infty}^{\infty} t^2 W_w(t)dt$$

W_w is normalized, so that

$$\int_{-\infty}^{\infty} W_w(t)dt=1.$$

M_2 (customarily given in mrads^2) has to be given in $(\Delta x)^2$ units.

A separate subroutine performs a slit-width correction on the data after the interpolation. The correction is made according to the equation

$$\tilde{I}(x)=\tilde{I}(x)- \frac{1}{2} M_2 \tilde{I}''(x)$$

Numerical evaluation of the second derivative of $\tilde{I}(x)$, based on a least-squared smoothing technique, is given in reference [1]. Let us assume that a series of k slit-width corrections apply, each in a different range of scattering angles. Let the second moment of the slit-width weighting functions in the range $W_1\Delta x \leq x \leq W_2\Delta x$ be $M_2(1)$, in the range $W_2\Delta x < x < W_3\Delta x$ be $M_2(2)$, etc. The last correction always applies in the range $W_k\Delta x \leq x < N\Delta x$. The following diagram 2 (Diagram 2 is presented on Page I) illustrates these corrections.

There are k cards for slit-width correction, one card for each range of scattering angles for which $M_2(i)$, i=1, 2, . . . k, is constant. These cards follow the first card of the program. These k cards are followed by a blank card. A blank card must be inserted, whether the data are corrected for the slit-width effect or not. Each one of these

k cards contain two numbers: W_j and $M_2(j)$, j=1, 2, . . ., k. They are punched FORMAT I10, F10.5.

A word of warning: It is advisable not to correct data for the slit-width effect in the steep part of the scattering curve, since the second derivative might be very large. It is recommended to apply the slit-width correction only at scattering angles in excess of 5 milli-radians.

The program user has four directive parameters which control the computations for the slit-length effects. These four parameters are punched on a single card, FORMAT 4I10. These are: MD, IDC, INV, MM. MD is an integer which must have one of the values 1, 2, 3, or 4. This integer controls the form of the weighting function to be employed in computations. These are, in order indicated by the value given to MD: 1) Constant weighting function; 2) Gaussian weighting function; 3) Data weighting function and; 4) Trapezoid weighting function. According to the form of the weighting function, additional information must be supplied, which describes the shape and form of this function, as follows:

MD=1 No additional input data are needed

MD=2 The weight function is defined as

$$W_i = \exp - [(i\Delta)^2/G]$$

Δ is the interval between successive values of W and $G/2$ is its variance. Δ is determined in such a way so that a table of 21 values of W are computed,

$$W_i = \exp -(0.01i^2);\ i=0, 1, \ldots, 20.$$

Thus, Δ, the interval between two successive values of W, is equal to 0.1 $\sqrt{G}$.

W is cut-off when it is less than e^{-4}. A single card giving G, FORMAT F10.2, must be inserted.

MD=3: The data weighting function is given by a set of values W(R). R should be equally spaced, but the spacing can be larger than the spacing of x in $\tilde{I}(x)$. R is given in Δx units. W can be given in arbitrary units. A set of cards giving the pairs of R and W, one pair per card, must be inserted, FORMAT F9.3, F11.5. As many cards as needed can be inserted. Last card must be blank.

MD=4: A single card giving S_1 and S_2 must be inserted (FORMAT 2F10.2). S_1 must be smaller than S_2. (The trapezoid is defined so that W=1 for $r<S_1$ and W=0 for $r<S_2$, and it is symmetric around r=0) (illustrated on page 33). S_1 and S_2 are in Δx units.

IDC, which, if different from zero, causes the computer to print out the values of the integrals

$$h_i = \int_{i\Delta x}^{N\Delta x} \frac{xI(x)dx}{(x^2-(i\Delta x)^2)^{1/2}}$$

for all values of i from m to N.

INV, which, if different from zero, causes the input vector $\tilde{I}$ to be recomputed from the output vector I (inversion of calculations).

MM, which, if different from zero, causes the integrals h_i to be calculated by the lower-order approximation (see Chapter 4 for details).

The input vector $\tilde{I}$ is read after all the above instructions were supplied. For MD=1 the input vector follows immediately the blank

card, which terminates the information needed for the slit-width correction.

For MD≠1, the input vector follows cards which characterize the particular weighting function employed in calculations.

The input vector $\tilde{I}$ is represented by a series of real numbers, eight numbers per card, FORMAT 8F10.0. The components of the input vector $\tilde{I}$ are $\tilde{I}_1$, $\tilde{I}_2$, ..., $\tilde{I}_{N_o}$. After N_o values of $\tilde{I}$ are fed into the machine, the computations will start, even if there are more data cards to be read. At the conclusion of the computations, new sets of data cards will be read, and new computations will be initiated. The program can handle more than one set of data, however, the instructions and specifications must remain the same. The computations are stopped when a blank card is read for $\tilde{I}_1$.

The instructions for the input parameters and variables are summarized in Table 1.

C. Brief Description of the Main Program and its Subroutines

The main program reads all the necessary parameters and variables which control the computation, followed by the input data. The input data are first shifted and interpolated as needed. The main program calls subroutine SWIDTH for the corrections required for the effects of the variable widths of the collimating slits. The calculations performed by the subroutine follow the method of Taylor and Schmidt [1]. Next, subroutines CGEN and BGEN are called to calculate the coefficients $\gamma_{i,j}$ and $b_{i,j}$ in the set of linear equations

$$\sum_{j=i}^{N} \gamma_{ij} \tilde{I}_j = \sum_{j=i}^{N} b_{ij} I_j$$

or, in matrix form, $\underset{\sim}{C}\underset{\sim}{I}=\underset{\sim}{B}\underset{\sim}{I}$. The main program performs an inversion of the matrix equation to compute $\underset{\sim}{I}$. The main program also prints out the results and performs the back-check (optional). Subroutine BGEN calls subroutine WGHT23 if MD=2 or 3, or calls subroutine WHGT4 if MD=4. The subroutines WGHT23 and WGHT4 calculate K(u,x), the kernel of the integral eq. (3), employing the selected weighting function. These calculations are performed for each pair of indices (i,j) (in notation of eq. (3), $u=i\Delta x$, $x=j\Delta x$, and $m<(i,j)<N$, where $m\Delta x$ is the lowest angle at which measurements were taken).

In addition, the program includes two function subroutines, DGEN and CGEN. These subroutines transform the coefficients $b_{i,j}$, calculated for the rectangle rule approximation (for which I(x) is constant in each integral and interpolates one end point) into the coefficients which obtain when I(x) is replaced by certain interpolating polynomials. This procedure is explained in detail in Chapters 3 and 4.

D. Comments on Units Employed

$\underset{\sim}{I}$, $\underset{\sim}{\tilde{I}}$, and $\underset{\sim}{W}$ are given in arbitrary units (usually as counts per seconds). The coordinates of $\underset{\sim}{\tilde{I}}$ and of $\underset{\sim}{W}$ are given in units of Δx, where $\Delta x=(\Theta_{i+1}-\Theta_i)$ and Θ_i is the scattering angle at the point i. If Δx is equal to s mrads, all other linear coordinates must be given in terms of multiples of s. No explicit value for Δx is needed, since in machine calculations, $\Delta x=1$. The variance in the gaussian distribution and M_2, the second moment of the width weighting function are in $(\Delta x)^2$ units. For example, if M_2 is measured as 0.15 mrads^2, and Δx is equal to 0.2 mrads, then M_2, in $(\Delta x)^2$ units, is equal to 3.75.

Occasionally, the unsmearing of data might be restricted to larger scattering angles, or the calculations for larger scattering angles are to be performed separately from the calculations at very low scattering angles. Suppose that the calculations start with the scattering angles of $p\Delta x$, at which the spacing between the consecutive scattering angles is equal to $\ell\Delta x$. In this case, ISTART=p, and no interpolation of input data is needed. All the linear coordinates employed will be in units of $\ell\Delta x$ instead of in units of Δx. In machine calculations, $\ell\Delta x=1$. Similarly, M_2, the second moment of the slit-width weighting function, will be in $(\ell\Delta x)^2$ units.

If the scattering intensities, starting at certain angle $p\Delta x$, are much smaller than the intensities at the lowest measured angles, then the unsmearing of data for very small scattering angles ($\Theta<p\Delta x$) might be performed on a restricted range of the scattering angles, terminated at the angle $p\Delta x$. This artificial cutting off the calculations generally will not affect the pattern of the scattering curve, that is the relative positions and intensities at various maxima and minima. In this case, the computations should be performed separately for very low scattering angles ($\Theta<p\Delta x$) and separately for larger scattering angles ($\Theta>p\Delta x$). This procedure might be preferable if the computer time is of primary concern. By performing calculations separately for different ranges of the scattering angles, N, the total number of the input data, is reduced. Since the processor time involved is roughly proportional to N^2, considerable computer time can be saved.

CHAPTER 3

1. THEORETICAL BACKGROUND

The equation which relates the experimentally determined intensity of scattered rays $\tilde{I}$ to the true scattering intensity I is given by eq. (2). In eq. (2), W may be given either in analytic form or as a set of discrete values. Eq. (2) is transformed as follows: both sides of eq. (2) are multiplied by $x/(x^2-u^2)^{1/2}$ and integrated over x from u to ∞. After exchanging the orders of integration one obtains eq. (3). The total number of experimental data for I(x) is N, which are given at equally spaced intervals Δx. We assume that, if $\tilde{I}(x)$ is sufficiently small when $x>N\Delta x$, so is I(x). Therefore, both the upper limits of integrals in eq. (3) are replaced by $N\Delta x$. Both $\tilde{I}(x)$ and I(x) represent a discrete set of function values $\tilde{I}_i$ and I_i, respectively, with i = 0, 1, ..., N. Thus, i=0 corresponds to u=0, i=N corresponds to $u=N\Delta x$. Actually, the calculations seldom start with u=0 but from a small angle equal to $m\Delta x$, m>0, since $\tilde{I}(x=0)$ is not known, unless it is obtained by extrapolation of data to zero angle.

The l.h.s. of eq. (3) can be written as a set of N linear equations:

$$h_i = \int_{i\Delta x}^{N\Delta x} \tilde{I}(x)G(u,x)dx \cong \sum_{j=i}^{N} \gamma_{i,j}\tilde{I}_j ; \tilde{I}_j=0 \text{ for } j>N, \qquad (5)$$

where

$$G(u,x) = \frac{x}{(x^2-u^2)^{1/2}}; \quad u=i\Delta x,\ i=m,\ m+1,\ \ldots,\ N.$$

In matrix form, $h=A\tilde{I}$. is an upper triangular matrix with entries coefficients γ_{ij}. The calculations of h_i, though they happen to represent improper integrals, are straight-forward and will be described in Chapter 4. Suffice here to say that in each summand of eq. (5), $\tilde{I}(x)$ is expanded in a Taylor series around $x=j\Delta x$, retaining only the first three terms of the expansion.

The right hand side of eq. (3) is expanded in terms of the discrete values I_i as

$$\int_{i\Delta x}^{N\Delta x} I(x)K(u,x)dx \tilde{=} \sum_{j=i}^{N} b_{ij}I_j;\ I_j=0 \text{ for } j>N \qquad (6)$$

b_{ij} are coefficients to be determined. They depend on the weighting function W. Therefore, from (5) and (6), I, which is the N-dimensional solution vector, is given in matrix notation by

$$I=B^{-1}h$$

B^{-1} is the inverse of the matrix B whose elements are the sets of the coefficents b_{ij} of eq. (6), and h is an N-dimensional vector with components h_i, eq. (5). Since the matrix B is an upper triangular one, its inversion can be performed rapidly by a back substitution, starting with the $b_{N,N}$ element of B, as follows:

$$\begin{aligned} I_N &= h_N/b_{N,N} \\ I_i &= (h_i - \sum_{j=i+1}^{N} b_{ij}I_j)/b_{ii},\ i=N-1,\ N-2,\ \ldots,\ m. \end{aligned} \qquad (7)$$

The method employed in this work for solving eq. (6) is based on a replacement of the function I(x) by an interpolating polynomial having

the values $I_k = I(x_k)$ on a set of points x_k.

This is a commonly used procedure employed in conjunction with multi-step methods which are based on interpolation. For a multi-step method of order ℓ, the integral in eq. (5) is replaced by a repeated equally spaced quadrature rule defined on $\ell+1$ points. Thus, eq. (3) is written as a system of (N-i+1) simultaneous equations

$$h_i = \sum_{j=i}^{N} \int_{j\Delta x}^{(j+1)\Delta x} I(x)K(i\Delta x,x)dx \qquad (8)$$

$I(x)$ is replaced by an interpolating polynomial having the values $I_k \equiv I(x_k)$ on a set of points x_k. In forward interpolation of multiplicity ℓ, the interpolating polynomial interpolates $I(x)$ on points x_k, x_{k+1}, ..., $x_{k+\ell}$. In general, the interpolating polynomial $P_\ell(x)$ of degree ℓ agrees with $I(x)$ and $\ell+1$ suitably chosen points, which can be located to the right, to the left or centered around the given point x_k, representing forward, backward or central interpolations, respectively. The integral of $I(x)K(i\Delta x,x)$ over the interval $[j\Delta x, (j+1)\Delta x]$ is then replaced by the integral of $P_k(x)K(i\Delta x,x)$ over the same range. We will consider here only two cases, $\ell=1$, and $\ell=2$.

First, assume $\ell=1$. In the simplest case, which we call the "rectangle rule" approximation, $I(x)$ in the r.h.s. of eq. (7) is replaced by I_j. In this case,

$$b_{ij} = \int_{j\Delta x}^{(j+1)\Delta x} K(i\Delta x,x)dx.$$

An improvement over the rectangle rule approximation for a single-step method (ℓ=1) would be, if I(x) in the integrals of eq. (8) are replaced by the following averages (trapezoid rule approximation):

$$I(x) = \frac{1}{2}(I_j+I_{j+1}) \text{ for } j\Delta x \leq x \leq (j+1)\Delta x. \tag{9}$$

In this case, the elements of the matrix $\mathbf{B}$ are:

$$b_{i,i} = \int_{i\Delta x}^{(i+1)\Delta x} K(i\Delta x,x)dx \qquad (j=i)$$

$$b_{i,j} = \int_{(j-1)\Delta x}^{(j+1)\Delta x} K(i\Delta x,x)dx \qquad (j>i) \tag{10}$$

$$b_{i,j}=0. \qquad (j<i)$$

The question is: Is the trapezoid rule approximation (eq. (9)) really an improvement over the rectangle rule approximation? Theoretically, the answer should be yes. Under the rectangle rule approximation, I(x) is assumed to be constant within each strip. Under a trapezoid rule, I(x) is assumed to be a linear function of x in each of the integrals of the r.h.s. of eq. (8), and then replaced by its value at the middle points of these integrals. Nevertheless, an application of the trapezoid rule might lead to even-odd oscillations in the computed I_i. These oscillations are associated with the presence of weak numerical instabilities. The weak instabilities, in contradistinction to strong numerical instabilities, can be greatly reduced by either adopting certain averaging procedures or by a smoothing method, and will be discussed later.

The investigations of the numerical instabilities, associated with the employment of various quadrature rules can be greatly facilitated by

examination of the elements of the matrix $\mathbf{B}^{-1}$. We first demonstrate the possibility of the presence of even-odd oscillations in the computed values of I_i under the trapezoid rule approximation, and the relative absence of these oscillations under the rectangle rule approximation. For simplicity, assume that W, the weighting function, is a constant, equal to $W=2/\pi$, so that $K(u,x)=2x$. Employing the rectangle rule approximation, that is, assuming that $I(x)=I_j$ in each of the integrals of the r.h.s. of eq. (8), one obtains from eq. (8),

$$b_{i,j}=2j+1 \qquad j\geq i$$
$$b_{i,j}=0 \qquad j<i \tag{11}$$

$\beta_{i,j}$, which are the elements of the matrix $\mathbf{B}^{-1}$ are found to be

$$\beta_{i,i} = \frac{1}{2i+1}$$

$$\beta_{i,i+1} = -\frac{1}{2i+1}$$

$$\beta_{i,j} = 0 \text{ for } j\geq i+2$$

Since $\mathbf{I}=\mathbf{B}^{-1}\mathbf{h}$, the computed I_i are seen to depend on h_i and h_{i+1} only. Thus, the rectangle-rule approximation will not introduce spurious short-range maxima and minima into the computed I(x) vs. x curves, if the original experimental $\tilde{I}(x)$ curves were reasonably smooth.

Next, consider the trapezoid rule approximation, eq. (9). Now,

$$b_{i,i} = \frac{1}{2}(2j+1) \qquad j=i$$

$$b_{i,j} = 2j \qquad j>i \qquad (12)$$

$$b_{i,j} = 0 \qquad j<i$$

For large i, the elements of the matrix B^{-1} are approximately,

$$\beta_{i,j} = \frac{4}{2i+1} \quad \text{for } j=i+2,\ i+4,\ i+6,\ \ldots$$

$$\beta_{i,j} = -\frac{4}{2i+1} \quad \text{for } j>i+1,\ i+3,\ i+5,\ \ldots \qquad (13)$$

$$\beta_{i,j} = \frac{2}{2i+1} \quad \text{for } j=i.$$

Thus,

$$I_i \tilde{=} h_i\beta_{i,i} + \frac{4}{2i+1}\left(-h_{i,i+1}+h_{i,i+2}-h_{i,i+3}+\ldots\right) \qquad (14)$$

and the possibility of even-odd oscillations in the computed I_i is self-evident.

Before getting into discussions of the methods that might partially eliminate these oscillations, let us next consider the case when I(x), eq. (8) is replaced by an interpolating polynomial having values on a set of points x_k. Let Q(x) be the polynomial which approximates and replaces I(x) in eq. (8).

This polynomial can be expressed in terms of forward, central or backward differences. Assume $\ell=2$. In central difference interpolation, the interpolating points are x_j, x_{j-1} and x_{j+1}. In backward interpolation, they are x_j, x_{j-1} and x_{j-2}. In forward interpolation, they are x_j, x_{j+1} and x_{j+2}.

Consider central difference interpolation (for $\ell=2$, this method is equivalent to the extended Simpson rule for numerical integration). Consider the case of W=constant=$2/\pi$. The interpolating polynomial Q(x) is given by

$$Q(x) = I_j + \frac{(I_{j-1} - 2I_j + I_{j+1})}{2(\Delta x)^2} (x - j\Delta x)^2 \qquad (15)$$

For i>>1, the elements of the ith row of the $\mathbb{B}$ matrix are, approximately

$$\begin{aligned} b_{i,i} &= i/3 \\ b_{i,j} &= 4j/3 \quad (j=i+1,\ i+3,\ i+5,\ \ldots) \\ b_{i,j} &= 2j/3 \quad (j=i+2,\ i+4,\ i+6,\ \ldots). \end{aligned} \qquad (16)$$

When this matrix is inverted, $\beta_{i,j}$, the elements of the inverted matrix, are approximately given by

$$\beta_{i,j} = \frac{(-1)^{j-i}}{i} \{(-2+\sqrt{3})^{j-i} + (-2-\sqrt{3})^{j-i}\}$$

Thus, the absolute values of $\beta_{i,j}$ are seen to grow exponentially as one moves away from the diagonal elements of the inverted $\mathbb{B}$ matrix. As a result, the numerical calculations are affected by strong numerical instability which renders the above method of constructing interpolating polynomial to be totally useless.

This strong instability is characterized as a pointwise phenomenon, as it involves error growth at fixed points as the interval between them decreases. In other words, the multi-step solution of eq. (8), when I(x) is replaced by the above given interpolating polynomial, diverges as $\Delta x \to 0$ [13].

The question is, how one constructs an interpolating polynomial Q(x) that will lead to numerically stable (or convergent) method for

calculating the solution vector]? This question was answered in detail in reference [6]. Suffice here to say that, if the coefficients of I_k in the interpolating polynomial are α_k, so that I(x) in r.h.s. of eq. (8) is replaced by

$$\alpha_i I_i + \alpha_{i+1} I_{i+1} + \ldots + \alpha_{i+\ell} I_{i+\ell}$$

then the moduli of all roots of the characteristic polynomial

$$\phi(\eta) = \alpha_i \eta^{\ell} + \alpha_{i+1} \eta^{\ell-1} + \ldots + \alpha_{i+\ell} = 0 \qquad (18)$$

must not exceed unity, as a condition for convergence of the multi-step method of multiplicity ℓ[1]. In the above example, in which central difference method was employed for the construction of the interpolating polynomial, the characteristic polynomial $\phi(\eta)$ is given by

$$\phi(\eta) = 1/3\eta^2 + 4/3\eta + 1/3$$

and the root condition for numerical stability is not satisfied. In reference [6] it was shown that, if instead of central differences, one employs forward differences, a convergent method can be obtained. (In multi-step method, employment of forward differences is a necessary, but not sufficient condition for numerical stability.) The interpolating points for $\ell=2$ are j, j+1 and j+2. The root-determining equation is

$$\phi(\eta) = 7/3\eta^2 - 2/3\eta + 1/3 = 0. \qquad (19)$$

Since the moduli of the roots of eq. (19) are less than 1, the method is numerically stable.

---

[1]This is a very serious restriction. As a matter of fact, it is very hard to construct an interpolating polynomial which will satisfy the root condition, when $\ell>2$.

For i>>1, the elements of the ith row of the $\mathcal{B}$ matrix (for constant W) are:

$$b_{i,i} = 7i/3$$

$$b_{i,j} = -2j/3 \qquad (j=i+1,\ i+3,\ i+5,\ \ldots)$$

$$b_{i,j} = 2j/3 \qquad (j=i+2,\ i+4,\ i+6,\ \ldots)\ .$$

We designate the above-described method of constructing interpolating polynomial for I(x) as the Nÿstrom algorithm [12] for the following reason:

In analogy to the Nÿstrom method for constructing interpolating polynomial for solving difference equations of first order, this polynomial is based on forward interpolation and is applied to the midpoint of each of the summands in the quadrature rule approximation of the integral equation (because it is applied to the midpoint, the coefficient of ∇I_k is zero, thus insuring numerical stability).

In our computer program, this algorithm is employed in addition to the rectangle rule and trapezoid rule approximations for replacing I(x), eq. (6) by a set of discrete values I_k.

On the basis of the above discussion on condition for numerical stability, it becomes clear why the one-step methods are always numerically stable: for the rectangle rule approximation, the root-determining characteristic polynomial $\phi(\eta)$ is of zero order. This method is therefore the most stable numerically. On the other hand, the trapezoid rule method implies that the characteristic polynomial is of the first order, with root equal to -1. This represents the border-line case for numerical stability.

All these considerations apply strictly to the case of W=constant. For a non-constant W we found that the stability conditions are slightly more relaxed. Specifically, for W represented by a trapezoid, it is easy to show that the absolute values of the coefficients $b_{i,j}$ decrease as j increases as compared with the case of constant W. This fact is of importance, particularly with conjunction with the trapezoid rule method, as it implies that a non-constant W tends to eliminate some of the even-odd oscillations in the computed I(x), as compared with the case of W=constant.

As expected, the Nÿstrom algorithm leads to results that are relatively free of even-odd oscillations. This is because $\beta_{i,j}$, the i,j element of B^{-1} matrix, is approximately proportional to $\frac{1}{2}|\lambda|^{j-i}$ where $|\lambda|$ is the modulus of the largest root of the characteristic polynomial $\phi(\eta)$. Since $\lambda<<1$, the matrix elements β_{ij} are seen to decrease rapidly as one moves, in a given row, away from the diagonal elements of the matrix B^{-1}.

As demonstrated above, the solution - vector I can display undesirable even-odd oscillations, particularly when $|\lambda|$, the modulus of the root of the characteristic polynomial, is close to one. This is particularly the case when trapezoid rule is applied to the solution of eq. (8). These oscillations can be greatly reduced by smoothing the results as developed in reference [6]. In this method, one computes first the solution vector $I=I^{(o)}$ from

$$I^{o} = B_{o}^{-1}h \tag{20}$$

B_o is the matrix of $b_{i,j}$, calculated from eq. (7,9, and 10). The "corrected" solution vector $I^{(1)}$ is obtained by calculating the second

order differences $\delta^2 I_i^{(o)} = I_{i-1}^{(o)} - 2I_i^{(o)} + I_{i+1}^{(o)}$ for each element $I_i^{(o)}$ of the solution vector $I^{(o)}$, and adding $\gamma\delta^2 I_i^{(o)}$ to $I_i^{(o)}$. For γ, a value of 1/6 was adopted. In reference [8], γ=1/4. However, we found, on the basis of numerous sample calculations, that the value γ=1/6 is preferable. This smoothing procedure is repeated, starting with the solution vector $I^{(1)}$, to obtain a solution vector $I^{(2)}$. Since the computations of the smoothed solution vectors $I^{(1)}$ and $I^{(2)}$ are very rapid, they are both included in the computer program.

In the final print-out, we present solutions for the scattered intensity, based on the rectangle rule approximation and on the trapezoid rule approximations (with and without smoothing). This is done for the following practical purpose: the user of the computer program should recognize that some spurious small peaks in the calculated I(x) curves should not be given any physical interpretation. Rather, these peaks, or oscillations, in the computed I(x) curves result from the presence of weak numerical instabilities. In order to be sure whether maxima and minima in the calculated I(x) curves have physical significance, one should examine the I(x) curves obtained from the application of the rectangle rule approximation. In the rectangle rule solution, weak numerical instabilities, resulting in even-odd oscillations, are usually absent, provided that the experimental $\tilde{I}(x)$ vs. x curves are also free of even-odd oscillations (which might come from noise effects or from errors in measurements). Otherwise, even with the rectangle rule approximation, these oscillations will be magnified in the I(x) vs. x curves. It is, therefore, very important that the experimentally

determined I(x) be smoothed before it is read into the computer. This is true in particular with respect to the trapezoid rule approximation.

The solution based on the Nÿstrom algorithm (multiple step method of multiplicity 2, with forward interpolation) is more accurate than the rectangle-rule solution, and is relatively free of even-odd oscillations, so that corrections based on smoothing the results need not be applied to this solution. The solution based on this method should be employed for display of desmeared intensities.

The various algorithms and their effect on the calculated I(x) are discussed in Chapter 6, where sample calculations employing various algorithms are presented and discussed.

CHAPTER 4

1. DESCRIPTION OF CALCULATIONS

A. Preparation of Data which Characterize the Slit Weighting Function

The following slit weighting functions are employed in this program: 1) Weighting function is a constant; 2) Gaussian Function; 3) Data Function; and 4) Trapezoid Function.

For Gaussian distribution, W(r) is computed at equal intervals equal to Δ. Thus, the program computes a set of discrete values:

$$W_i = \exp(-i^2\Delta^2/G)$$

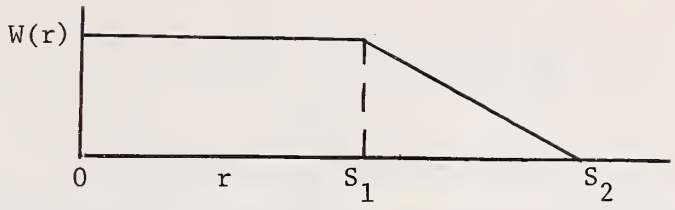
G is read from the input data card, and Δ is arbitrarily chosen to equal $0.1\sqrt{G}$. The subscript i assumes values of 0, 1, ..., 20.

W(r) is represented by data function. A set of cards giving the various pairs of W(r) and r are read into the computer. The data are represented as $r_1(=0)$, r_2, r_3, ..., r_m; $W(r_1)$, $W(r_2)$, $W(r_3)$, ..., $W(r_m)$. $W(r_i)$ are given at equally spaced intervals. The first pair of data are stored as r_2 and $W(r_2)$, respectively. r_1 is set as equal to zero, and $W(r_1)$ is computed according to

$$W(0)=W(r_1)=W(r_2)+[W(r_2)-W(r_3)]r/(r_3-r_2).$$

There are m pairs of data. If m<N, the weighting function W(r) for $r>r_m$ is given a constant value which is equal to $W(r_m)$. Thus, $W(r_k) = W(r_m)$ for $m<k<N$.

W(r) is represented by a trapezoid, as follows:



$W(r)=1 \qquad r<S_1$

$W(r)=\dfrac{S_2-r}{S_2-S_1} \qquad S_1 \leq r \leq S_2$

$W(r)=0 \qquad r>S_2$

$W(-r)=W(r)$.

B. Evaluation of the Integral

$$h(u) = \int_u^\infty \frac{x\tilde{I}(x)}{(x^2-u^2)^{1/2}}dx = \int_{i\Delta x}^{N\Delta x} \frac{x\tilde{I}(x)}{[x^2-(i\Delta x)^2]^{1/2}}dx \qquad (21)$$

since $\tilde{I}(x)=0$ for $x>N\Delta x$. The integral $h(u) = h(i\Delta x)$ is computed for each value of i by the subroutine CGEN which is entered with a given value of i. To evaluate eq. (21) we subdivide the integral into N-i-1 integrals as follows:

$$h_i = \sum_{j=i}^{N} \int_{j\Delta x}^{(j-i)\Delta x} \frac{x\tilde{I}(x)}{[x^2-(i\Delta x)^2]^{1/2}}\,dx \qquad (22)$$

I(x) is expanded in a Taylor series around $x = j\Delta x$:

$$h_i = \sum_{j=i}^{N} \int_{j\Delta x}^{(j+1)\Delta x} \left\{\tilde{I}_j + [x-(j+1)\Delta x]\,\frac{\tilde{I}_{j+1}-\tilde{I}_j}{\Delta x} + \frac{1}{2}\,[x-(i+1)\Delta x]^2 \right.$$

$$\left. x\,\frac{(\tilde{I}_{j+1}-2\tilde{I}_j+\tilde{I}_{j-1})}{(\Delta x)^2}\right\}\frac{x}{[x^2-(i\Delta x)^2]^{1/2}}\,dx \qquad (23)$$

$$h_i = \sum_{j=i}^{N} \gamma_{ij} \tilde{I}_j ; \quad \tilde{I}_j = 0 \text{ for } j>N. \tag{24}$$

In eq. (24) and in following equations, $\tilde{I}_j = \tilde{I}(j\Delta x)$, etc. $\gamma_{i,j}$ form an upper triangular matrix C. The matrix elements $\gamma_{i,j}$ have different representations depending on the following three cases: (1) j-i=0, (2) j-i even and (3) j-i odd. In addition, in order to close the system one has to consider, separately, the cases of j=N and j=N-1.

Sometimes the high accuracy in calculating h_i is not required. One can simplify calculations and save computer time by retaining, in the r.h.s. of eq. (23), only the first term in the Taylor expansion of I(x). This option is controlled by the parameter MM, read with the data cards. Since in this case the results are simple, we spell them here. They might be of use for a fast check of the results:

$\gamma_{i,j}=1/2\ \{[(j+1)^2-i^2]^{1/2} - [(j-1)^2 - i^2)^{1/2}\}\Delta x$ for N-1>j>i

$\gamma_{i,j}=1/2\ (2i+1)^{1/2}\Delta x$ for j=i

$\gamma_{i,j}=0$ for j<i.

For j=N, $\gamma_{i,j}=1/2\ \{[(j^2-i^2)]^{1/2} - [(j-1)^2-i^2]^{1/2}\}\Delta x$.

C. Evaluation of the Kernel of the Integral Equation for Various Weighting Functions

1. Preliminary Consideration

We wish to determine the coefficients $b_{i,j}$ in the set of simultaneous equations

$$\sum_{j=i}^{N} b_{i,j} I_j = h_i \tag{25}$$

given that

$$h_i = \int_{i\Delta x}^{N\Delta x} I(x)K(i\Delta x,x)dx \tag{26}$$

$$K(i\Delta x,x) = 2x \int_o^{[x^2-(i\Delta x)^2]^{1/2}} \frac{W(t)}{[x^2-(i\Delta x)^2-t^2]^{1/2}}\, dt. \tag{27}$$

The integral of eq. (27) is replaced by the sum of integrals

$$h_i = \sum_{j=i}^{N} \int_{j\Delta x}^{(j+1)\Delta x} I(x)K(i\Delta x,x)dx \tag{28}$$

We assume that in eq. (28), I(x) can be replaced by its average value within each range and taken outside the integral sign provided that I(x) is continuous on $[j\Delta x, (j+1)\Delta x]$.

Define

$$\bar{b}_{i,j} = \int_{(j^2-i^2)^{1/2}\Delta x}^{[(j+1)^2-i^2]^{1/2}\Delta x} dtt \int_o^t \frac{W(r)}{(t^2-r^2)^{1/2}}\, dr \tag{29}$$

$\bar{b}_{i,j}$ are identical to $b_{i,j}$ only when I(x), eq. (28), is equal to I_j. Otherwise, $b_{i,j}$ will be computed from $\bar{b}_{i,j}$, according to the algorithm employed in replacing I(x), eq. (28), by a set of discrete values I_k. Introduce

$$R_1 = (j^2-i^2)^{1/2}\Delta x, \quad R_2 = [(j+1)^2-i^2]^{1/2}\Delta x.$$

From eq. (29), after changing order of integration, we obtain

$$\bar{b}_{i,j} = \int_{R_1}^{R_2} drW(r) \int_{r}^{R_2} \frac{t}{(t^2-r^2)^{1/2}}\, dt + \int_{o}^{R_1} drW(r) \int_{R_1}^{R_2} \frac{t}{(t^2-r^2)^{1/2}}\, dt \qquad (30)$$

Integrate over t, to obtain

$$\bar{b}_{i,j} = \int_{o}^{R_2} (R_2{}^2-r^2)^{1/2} W(r)dr - \int_{o}^{R_1} (R_1{}^2-r^2)^{1/2} W(r)dr. \qquad (31)$$

Write

$$\bar{b}_{ij} = B_{i,j+1} - B_{i,j}. \qquad (32)$$

To compute $B_{i,j}$ the FORTRAN program calls a Function subprogram B(I,J) for each pair of values i and j. This procedure is employed in order to prevent storing large matrices. In another program which is to be employed when N<140, the elements $b_{i,j}$ are stored in a matrix array, and the FORTRAN program calls a subroutine B(I,J). The function B(I,J) or the subroutine B(I,J) calls other subroutines for evaluation of the integrals in eq. (31). The calculations for each pair of variables i and j are performed twice. The subroutine that computes eq. (31) is entered with a variable $R=R_2$ the first time, and with a variable $R=R_1$ the second time.

Both the Gaussian and the Data function representation for W(r) are treated in the same way, except that, for the Gaussian W(r), W(r) are first computed by the Main Program. The computation of $B_{i,j+1}$ and of $B_{i,j}$ for the Gaussian and Data function representations of W(r) are performed in the subroutine WGHT23. For the trapezoid W(r), these computations are performed in the subroutine WGHT4. For W(r) = constant, $\bar{b}_{i,j} = \frac{\pi}{4}(2j+1)$.

2. Computation of eq. (31)

a) W is given by a set of data. Subroutine WGHT23 is entered the first time with $R = [(j+1)^2-i^2]^{1/2}\Delta x$, the second time with $R = (j^2-i^2)^{1/2}\Delta x$. Since R is generally a non-integer, we locate an integer R, such that $R_\ell < R < R_{\ell+1}$.

Calculate W(R) by linear interpolation as follows:

$$W(R) = W(R_\ell) + \frac{W(R_{\ell+1})-W(R_\ell)}{R_{\ell+1}-R_\ell}(R-R_\ell) \qquad (r_m > R)$$

or

$$W(R) = W(r_m) + \frac{W(r_m)-W(r_{m-1})}{r_m - r_{m-1}}(R-r_m) \qquad (r_m \leq R). \tag{34}$$

$W(r_m)$ and r_m is the last pair of data read into the computer. Each of the integrals, eq. (31), is represented by a sum of smaller integrals. $B_{i,j+1}$ is written as

$$B_{i,j+1} = \sum_{k=1}^{\ell} \int_{r_{k-1}}^{r_k} (R_2{}^2-R^2)^{1/2}\, W(r)dr + \int_{r_\ell}^{R_2} (R_2{}^2-r^2)^{1/2} W(r)dr. \tag{35}$$

In each of the integrals in the summation, eq. (35), W(r) is approximated by a linear dependence on r within each strip (r_{k-1}, r_k). The individual integrals are expressed as follows:

$$\int_{r_{k-1}}^{r_k} (R_2{}^2-r^2)^{1/2} W(r)dr = \frac{W_k+W_{k-1}}{2} \int_{r_{k-1}}^{r_k} (R_2{}^2-r^2)^{1/2} dr + \tag{36}$$

$$\frac{W_k-W_{k-1}}{r_k-r_{k-1}} \int_{r_{k-1}}^{r_k} (R_2{}^2-r^2)^{1/2} \left(r - \frac{r_k+r_{k-1}}{2}\right) dr.$$

In eq. (36), $W_k=W(r_k)$, etc. In eq. (36) the last term is an improvement over the trapezoid rule approximation.

Performing the integrations and rearranging terms, one obtains

$$B_{i,j+i} = \frac{1}{2} \sum_{k=1}^{\ell-1} k(W_{k-1}-2W_k+W_{k+1}) \left\{ R_2^2 \sin^{-1}\left(\frac{k\Delta}{R_2}\right) + k\Delta[R_2^2-(k\Delta)^2]^{1/2} \right.$$

$$\left. + \frac{2}{3k\Delta}[R_2^2-(k\Delta)^2]^{3/2} \right\} + \left\{ W_{\ell-1} - \frac{\Delta}{R_2-\ell\Delta}(W_\ell-W_{R_2})-W_\ell \right\}$$

$$X \frac{\ell}{2} R_2^2 \sin^{-1}\left(\frac{\ell\Delta}{R_2}\right) + \ell\Delta[R_2^2-(\ell\Delta)^2]^{1/2} + \frac{2}{3\ell\Delta}[R_2^2-(\ell\Delta)^2]^{3/2}$$

$$+ \frac{\pi R_2^2}{8}\left\{ W_{R_2} + W_\ell - \frac{R_2+\ell\Delta}{R_2-\ell\Delta}(W_{R_2}-W_\ell) \right\}. \qquad (37)$$

In eq. (37), $r_k=k\Delta$, $\Delta=r_k-r_{k-1}$. In deriving eq. (37), use was made of the fact that $r_o=0$, by definition. In deriving the last two terms of eq. (37), $(\ell-1)W_\ell\Delta$ and $R_2W_\ell\Delta$ were approximated by $\ell W_\ell\Delta$. Similar results are obtained for R_1 replacing R_2. As a fast check, assume a constant $W(r) = W$. In this case, $\bar{b}_{i,j} = \frac{\pi}{4}(R_2^2-R_1^2)$, which also obtains directly from eq. (31).

b) Trapezoid Weighting Function. Let

$$W(t) = 1 \qquad (t<S_1)$$

$$W(t) = (S_2-t)/(S_2-S_1) \qquad (S_1<t<S_2) \qquad (38)$$

$$W(t) = 0 \qquad (t>S_2)$$

Substitute (39) into (30) to obtain

1) $R_2 > S_2$

$$B_{i,j+1} = S_1(R_2^2-S_1^2)^{1/2} + R_2^2 \sin^{-1}\left(\frac{S_1}{R_2}\right) + S_2(R_2^2 - S_2^2)^{1/2}$$

$$-S_1(R_2^2-S_1^2)^{1/2} + R_2\left[\sin^{-1}\left(\frac{S_2}{R_2}\right) - \sin^{-1}\left(\frac{S_1}{R_1}\right)\right]\frac{S_2}{S_2 - S_1} \qquad (39)$$

$$+ \frac{2}{3}\left[(R_2^2-S_2^2)^{3/2} - (R_2^2-S_1^2)^{3/2}\right]\frac{1}{S_2-S_1}$$

2) $S_1 < R_2 < S_2$

$$B_{i,j+1} = S_1(R_2^2 - S_1^2)^{1/2} + R_2^2 \sin^{-1}\left(\frac{S_1}{R_2}\right) + \frac{\pi}{2} R_2^2 - S_1(R_2^2-S_1^2)^{1/2}$$

$$- R_2^2 \sin^{-1}\left(\frac{S_1}{R_2}\right)\frac{S_2}{S_2 - S_1} - \frac{2}{3}\frac{(R_2^2-S_1^2)^{3/2}}{(S_2-S_1)} \qquad (40)$$

3) $R_2 < S_1$

$$B_{i,j+1} = \frac{\pi}{2} R_2. \qquad (41)$$

Similar expressions are obtained for $B_{i,j}$, with R_1 replacing R_2.

D. Calculations of $b_{i,j}$

For the rectangle rule approximation, $b_{i,j} = \bar{B}_{i,j}$. For the trapezoid rule approximation, we calculate first solution vector $I^{(o)}$ for the rectangle rule approximation,

$$I^{(o)} = B^{-1}h \qquad (42)$$

where B^{-1} is the inverse matrix of the elements $b_{i,j}$. Then,

$$I^{(\text{trapezoid})} = D^{-1}I^{(o)} \qquad (43)$$

the elements of the D matrix are

$$D_{i,i} = D_{i,i+1} = 1/2$$

$$D_{i,j} = 0 \quad j<i+1 \text{ and } j>i.$$

Therefore, $\mathbb{B} = \mathbb{D}\,\bar{\mathbb{B}}$. For the Nÿstrom algorithm (I(x) extrapolated forward), we employ the following approximate transformation: Form matrix $\mathbb{F}$ of elements $f_{i,j}$ as follows:

$$f_{i,i} = \frac{7}{3} b_{i,i}$$

$$f_{i,j} = \frac{2}{3} \bar{b}_{i,j} \quad (j=i+1,\ i+3,\ \ldots) \tag{44}$$

$$f_{i,j} = \frac{8}{3} \bar{b}_{i,j} \quad (j=i+2,\ i+4,\ \ldots).$$

Equations (45) are based on the following arguments: To insure numerical stability, the point of reference for interpolating polynomial must be located in the middle of the integrand. Therefore, eq. (28) is replaced by

$$h_i = \sum_{j=i,i+2,\ \ldots}^{N} \int_{(j-1)\Delta x}^{(j+1)\Delta x} I(x)K(i\Delta x,x)dx \tag{45}$$

Q(x), the interpolating polynomial which replaces I(x), is given by

$$Q(x) = I_j + (I_j - 2I_{j+1} + I_{j+2})\,(x-j\Delta x)^2/2(\Delta x)^2. \tag{46}$$

Therefore, the calculations of I_i must start with i=m+1, since with this algorithm, calculation of I_m will always be numerically unstable, as then the root condition (eq. 18) will not be satisfied. To close the system,

$$\begin{aligned} f_{i,j} &= 2/3\ \bar{b}_{i,j} && (j=N,\ j-i \text{ even}) \\ f_{i,j} &= 1/3\ \bar{b}_{i,j} && (j=N,\ j-i \text{ odd}) \\ f_{i,j} &= 2/3\ \bar{b}_{i,j} && (j=N-1,\ j-i \text{ even}) \end{aligned} \tag{47}$$

and of course,

$$b_{i,j} = f_{i,j},\ \mathbb{B} = \mathbb{F}.$$

Therefore,

$$\underset{\sim}{I}^{(\ddot{\text{Nystrom}})} \simeq \underset{\sim}{F}^{-1}\underset{\sim}{h}.$$

E. Solution Checking

In order to be able to make a direct check on the accuracy of the original calculations, we recalculated $\tilde{I}(x)$ by numerical integration, from

$$\tilde{I}_i = 2\int_{i\Delta x}^{N\Delta x} \frac{yI(y)W(y^2 - (i\Delta x)^2)^{1/2}}{[y^2 - (i\Delta x)^2]^{1/2}}\, dy = 2\int_{o}^{(N^2-i^2)^{1/2}\Delta x} W(t)(I[t^2-(i\Delta x)^2]^{1/2})dt. \tag{48}$$

$I([t^2-(i\Delta x)^2)]^{1/2})$ is calculated from the computed I by quadratic interpolation.

```
        CHAPTER 5
        FØRTRAN PRØGRAM
C       CALCULATIØN FØR SLIT CØRRECTIØN IN SMALL ANGLE X-RAY SCATTERING
C      PRØGRAM AVØIDS MATRIX STØRAGE
       DIMENSIØN XI(400),YI(400),XX(400),ZI(400),YK(400),YL(400),YY(400),
      1YM(400),U(400)
       DIMENSIØN LW(10),WM2(10)
       CØMMØN/BLØCK1/A,N,ISTART,NN,MD,MM
       CØMMØN/BLØCK2/RN(400),W(400),V(400),C(400),DL,G
       CØMMØN /BLØCK3/ KB,KL
       CØMMØN /BLØCK5/S1,S2
       DATA PI/3.1415926/
       DATA IPR,IRD/6,5/
C      READ DIRECTIVE INSTPUCTIØNS AND SET NUMBER ØF CØMPØNENTS IN INPUT VECTØR
       WRITE(IPR,1)
C     AT SØME PØINT THE INTERVAL BETWEEN INPUT DATA IS GREATER THAN AT
C     INITIAL ANGLES.THAT PØINT IS GIVEN BY 'IENTR'.THE NUMBER ØF SUBDI-
C     VISIØNS BETWEEN CØNSECUTIVE INPUT DATA FRØM THIS PØINT IS GIVEN BY
C     'NSUB'.
C     N IS THE NUMBER ØF CØMPØNENTS ØF THE INPUT VECTØR AFTFR INTERPØLA-
C     TIØN.NN IS THE NUMBER ØF INPUT DATA READ INTØ THE MACHINE.
C     ISTART IS THE RATIØ ØF THE LØWEST ANGLE AT WHICH MEASUREMENT IS
C     PERFØRMED TØ THE INTERVAL BETWEEN TWØ CØNSECUTIVE ENTRANCES.IT IS
C     GIVEN AS INTEGER
       READ(IRD,10) NN,ISTART,IENTR,NSUB
       SSUB=NSUB
10     FØRMAT (4I10)
1      FØRMAT(1H1 // 10X,15H SPECIFICATIØNS)
       N=NSUB*(NN-IENTR*ISTART)+NN*ISTART
        WRITE(IPR,30) N,NN,ISTART
30     FØRMAT(1X,23HN,TØTAL RANGE ØF DATA= ,I3,4X,24HNN,NUMBER ØF DATA RE
      1AD= ,I3,4X,24HISTART,STARTING PØINT,= I3)
       IF(IENTR.EQ.0) GØ TØ 36
       WRITE(IPR,35) IENTR,NSUB
35     FØRMAT(1X,33HENTRANCE PØINT FØR SUBDIVISIØN,= ,I3,4X,27HNUMBER ØF
      1SUBTABULATIØNS,= ,I3)
36      KØ=0
31     KØ=KØ+1
C     DATA FØR WIDTH CØRRECTIØN.WM2 IS THE SECØND MØMENT AND LW IS THE
C     FIRST PØINT AT WHICH THE WIDTH CØRRECTIØN APPLIES. SEVERAL WIDTH
C     CØRRECTIØNS CAN BE PERFØRMED IN SUCCESSIØN.
       READ(IRD,32) LW(KØ),WM2(KØ)
32     FØRMAT(I10,F10.5)
       IF(LW(KØ).EQ.0) GØ TØ 33
       WRITE(IPR,37) LW(KØ),WM2(KØ)
37     FØRMAT(5X,30HWIDTH CØRRECTIØN APPLIED FRØM ,I5,8X,15HSECØND MØMENT
      1= ,F10.5)
       GØ TØ 31
33     KØ=KØ-1
       WRITE(IPR,200)
38     FØRMAT(/)
       IØ=ISTART+1
       M=N-1
       N1=N+1
C       SELECT THE WEIGHTING FUNCTIØN
       READ(IRD,10) MD,IDC,INV,MM
        GØ TØ (40,60,100,230),MD
C       WEIGHTING FUNCTIØN IS CØNSTANT
40     WRITE(IPR,50)
50     FØRMAT(20X,31HWEIGHTING FUNCTIØN IS CØNSTANT //)
```

```
      GO TO 260
C      PREPARATIONS FOR GAUSSIAN WEIGHTING FUNCTION
C      DATA FOR GAUSSIAN REPRESENTATION OF WEIGHTING FUNCTION ARE TERMI-
C     NATED WHEN W(I) IS EQUAL TO 0.02
60    READ(IRD,70) G
70    FORMAT(2F10.2)
      DL=SQRT(0.01*G)
      DO 80 I=1,21
      EI=I
      RN(I)=(EI-1.0)*DL
80    W(I)=EXP(-RN(I)**2/G)
      WRITE(IPR,90) DL,G
90    FORMAT(4X,19HW IS GAUSSIAN, DL= ,F7.3,4H G= ,F7.1//)
      KB=1
      KL=21
      GO TO 260
C      PREPARATIONS OF DATA FOR THE DATA WEIGHTING FUNCTION
100   K=1
110   K=K+1
      READ(IRD,120) RN(K),W(K)
120   FORMAT (F9.3,F11.5)
      IF (RN(K).GE.0.00000001) GO TO 110
      IF (W(K).GE.0.00000001) GO TO 110
      DL=RN(K-1)-RN(K-2)
      EN=N
      IF (RN(2).EQ.0.) GO TO 130
C      FIND W(1) BY EXTRAPOLATION
      KB=1
      RN(1)=0.
      W(1)=W(2)+(W(2)-W(3))*RN(2)/(RN(3)-RN(2))
      GO TO 140
130   KB=2
140   IF (RN(K-1).LE.EN) GO TO 150
      KL=K-1
      GO TO 170
C      EXTEND THE WEIGHTING FUNCTION TO THE END OF THE RANGE OF COMPUTATIONS
150   KL=EN/DL+1.5
      DO 160 I=K,KL
      RN(I)=RN(I-1)+DL
160   W(I)=W(K-1)
C      PRINT OUT THE WEIGHTING FUNCTION
170   WRITE(IPR,180)
180   FORMAT (50X,23H W IS A DATA FUNCTION. //34X,13H GIVEN DATA..//41X,
     2 2H K,19X,2H R,19X,2H W//)
      DO 210 K=KB,KL
      WRITE(IPR,190) K,RN(K),W(K)
190   FORMAT (40X,I3,14X,F9.3,12X,F11.5/)
      SK=K/5
      XK=K
      IF (XK/5.0.NE.SK) GO TO 210
      WRITE(IPR,200)
200   FORMAT (/)
210     CONTINUE
      WRITE(IPR,220)
220   FORMAT (1H1)
      GO TO 260
C      PREPARATIONS FOR THE TRAPEZOID WEIGHTING FUNCTION
230   READ(IRD,70) S1,S2
      WRITE(IPR,240) S1,S2
240   FORMAT(4X,19HW IS TRAPEZOID,S1= ,F7.3,5H S2= ,F7.3//)
```

```
      DL=1.0
      DO 250 I=1,N1
250   RN(I)=I-1
C      READ IN INPUT VECTOR
C       IF BLANK,STOP THE CALCULATIONS
C       READ ONLY N COMPONENTS
260   READ(IRD,270) (XI(I),I=1,8)
270   FORMAT (8F10.0)
      IF (XI(1).LE.0.00000001) GO TO 800
      READ(IRD,270) (XI(I),I=9,NN)
C    SHIFT THE INPUT DATA BY ISTART
      DO 265 I1=1,NN
      I=NN+1-I1
      II=I+ISTART
265   XI(II)=XI(I)
C    DATA INTERPOLATION BY QUADRATIC FORWARD INTERPOLATION.IF IENTR IS
C    EQUAL TO ZERO,DO NOT INTERPOLATE
      IF(IENTR) 276,276,271
271   S=1.0/(1.0+SSUB)
      N0=NN+ISTART-2
      N2=NN+ISTART
      JENTR=IENTR
      K=JENTR
       DO 273 I=JENTR,N0
      XX(K)=XI(I)
      DO 272 J=1,NSUB
      T=J
      K=K+1
272   XX(K)=XI(I)+T*S*(XI(I+1)-XI(I))+(XI(I+2)-2.*XI(I+1)+XI(I))*T*S
     1*(T*S-1.)/2.0
273   K=K+1
      XX(K)=XI(N2-1)
      DO 274 J=1,NSUB
      T=J
      K=K+1
274   XX(K)=XI(N2-1)+T*S*(XI(N2)-XI(N2-1))
      XX(K+1)=XI(N2)
      DO 275 I=JENTR,N
275   XI(I)=XX(I)
276   CONTINUE
C    CORRECT INPUT DATA FOR WIDTH,IF NEEDED
      IF(KO.GE.1) CALL WIDTH(XI,LW,WM2,KO)
      DO 290 I=IO,N
      CALL CGEN (I)
C       MULTIPLY MATRIX C BY THE INPUT VECTOR
C       CALCULATE AND STORE AS V(I) THE ITH ROW OF MATRIX C
      C(I)=0.0
      DO 290 J=I,N
290   C(I)=C(I)+V(J)*XI(J)
C       MATRIX INVERSION BY BACK SUBSTITUTION,ZEROTH ORDER APPROXIMATION
310   DO 320 I=IO,N
320   YY(I)=C(I)
      A=1.0
      DO 340 K1=IO,N
      K=IO+N-K1
      DO 340 I=IO,K
      IF (I.EQ.K) GO TO 330
      YY(I)=YY(I)-YY(K)*B(I,K)/B(K,K)
      GO TO 340
330   YY(I)=YY(I)/B(I,I)
```

```
340   CONTINUE
C      OPTIONAL PRINT-OUT
      IF (IDC.EQ.0) GO TO 390
      IF(MM.NE.0) WRITE(IPR,380)
      WRITE(IPR,350)
350   FORMAT(///8X,11HC-VALUES.. ///)
      DO 360 I=IO,N
      II=I-1
360   WRITE(IPR,370) II,C(I)
370   FORMAT(15X,I4,E15.5)
380   FORMAT(8X,35HLOW ORDER APPROXIMATION IN C-VALUES/)
C      MATRIX INVERSION,TRAPEZOID RULE APPROXIMATION
390   DO 400 I=IO,N
400   YI(I)=YY(I)
      YI(N)=YY(N)/2.0
      DO 420 K1=IO,N
      K=IO+N-K1
      DO 420 I=IO,K
      IF (I.EQ.K) GO TO 410
      YI(I)=YI(I)-YI(K)*D(I,K)/D(K,K)
      GO TO 420
410   YI(I)=YI(I)/D(I,I)
420   CONTINUE
C      MATRIX INVERSION,NYSTROM ALGORITHM (FORWARD INTERPOLATION)
      A=0.
      DO 430 I=IO,N
430   YM(I)=C(I)
      YM(N)=YM(N)/2.0
      DO 450 K1=IO,N
      K=IO+N-K1
      DO 450 I=IO,K
      IF (I.EQ.K) GO TO 440
      YM(I)=YM(I)-YM(K)*F(I,K)/F(K,K)
      GO TO 450
440   YM(I)=YM(I)/F(I,I)
450   CONTINUE
C      SHIFT OUTPUT VECTOR YM BY ONE.SET YM(1) TO ZERO,SINCE IT CANNOT
C     BE COMPUTED WITH THIS ALGORITHM.
      DO 451 I1=IO,N
      I=IO+N-I1
451   YM(I+1)=YM(I)
      YM(IO)=0.
C      APPLY LOCAL PERTURBATIONS TO REDUCE EVEN-ODD OSCILLATIONS
470   DO 480 I=2,M
480   YK(I)=(YI(I-1)-2.0*YI(I)+YI(I+1))/6.0
      YK(1)=0.0
      DO 490 I=1,M
490   YK(I)=YI(I)+YK(I)
      DO 500 I=2,M
500   YL(I)=(YK(I-1)-2.0*YK(I)+YK(I+1))/6.0
      YL(1)=0.0
      DO 510 I=1,M
510   YL(I)=YK(I)+YL(I)
C      PRINT OUT RESULTS
540   WRITE(IPR,545)
545   FORMAT(// 8X,12HINPUT VECTOR,2X, 14HRECTANGLE RULE,2X,9HTRAPEZOID,
     12X,15HTRAPEZ.SMOOTHED,2X,14HYK(I) SMOOTHED,4X,21HNYSTROM INTERPOLA
     2TION)
      WRITE(IPR,550)
550   FORMAT(/3X,1HI,6X,5HXI(I),10X,5HYY(I),9X,5HYI(I),10X,5HYK(I),10X,
```

```
     15HYL(I),10X,5HYM(I)/)
      J=0
      DO 560 I=IO,N
      II=I-1
      J=J+1
      IF (J.LE.10) GO TO 560
      J=1
      WRITE(IPR,38)
560   WRITE(IPR,580) II,XI(I),YY(I),YI(I),YK(I),YL(I),YM(I)
580   FORMAT (I4,6E15.6)
C      IF NO BACK CHECK,FIND OUT WHETHER THERE ARE MORE DATA TO BE READ
C       IF NOT,GO TO STOP
      IF (INV.EQ.0) GO TO 260
C      PREPARATIONS FOR THE BACK CHECK
C      THE OUTPUT VECTOR TO BE CHECKED IS PUT INTO YI ARRAY.
C      INVERSE CALCULATIONS BY EXTENDED SIMPSON QUADRATURE
      IF (MD.EQ.1) GO TO 730
      IF (MD.EQ.4) GO TO 760
C      DATA PREPARATIONS FOR THE DATA WEIGHTING FUNCTION FOR INVERSE CHECK
      EN=N
      KL=EN/DL+1.5
      DO 610 I=2,N
      S=I
      DO 600 K=2,KL
      II=RN(K)
      IF (I.LE.II) GO TO 610
600   CONTINUE
610   U(I)=(W(K)-W(K-1))*(S-RN(K-1))/DL+W(K-1)
      U(1)=W(1)
620   DO 630 I=1,N1
630   RN(I)=I-1
C      DETERMINE OUTPUT VECTOR ON WHICH BACK-CHECKS ARE PERFORMED
      DO 590 I=1,N
C     DETERMINE OUTPUT VECTOR ON WHICH BACK CHECK IS PERFORMED.YI(I)
C      OF THE RIGHT HAND SIDE OF NEXT CARD CAN BE REPLACED BY YY(I),
C     YK(I),YL(I),OR YM(I).
590    YI(I)=YI(I)
      DO 680 I=IO,N
      S=I
      ZI(I)=W(1)*YI(I)/1.5
      DO 680 J=2,N1
      T=J
      R=SQRT((S-1.0)**2+(T-1.0)**2)
      R1=N1+1
      IF (R.GE.R1) GO TO 680
      DO 640 K=IO,N1
      IF (R.LT.RN(K)) GO TO 650
640   CONTINUE
      Y1=YI(N)+(YI(N)-YI(N-1))*(R-RN(N))/(RN(N)-RN(N-1))
      GO TO 660
650   Y1=YI(K-1)+(YI(K)-YI(K-1))*(R-RN(K-1))/(RN(K)-RN(K-1))+(YI(K+1)-2.
     20*YI(K)+YI(K-1))*0.5*(R-RN(K-1))*((R-RN(K-1))/(RN(K)-RN(K-1))-1.0)
     3/(RN(K)-RN(K-1))
660    IF (MOD(J,2).EQ.0) GO TO 670
      ZI(I)=ZI(I)+U(J)*Y1*4.0/3.0
      GO TO 680
670   ZI(I)=ZI(I)+U(J)*Y1*8.0/3.0
680   CONTINUE
C      PRINT OUT THE ORIGINAL AND THE RECONSTRUCTED OUTPUT VECTORS
      WRITE(IPR,690)
```

```
690   FØRMAT(1H1 // 20X,11HBACK CHECKS /20X,5HINPUT,5X,10HRECØMPUTED)
700   WRITE(IPR,710)
710   FØRMAT(//15X,27HI       XI(I)          ZI(I)/)
      J=0
      DØ 720 I=IØ,N
      II=I-1
      J=J+1
      IF (J.LE.10) GØ TØ 720
      J=1
      WRITE(IPR,38)
720   WRITE(IPR,725) II,XI(I),ZI(I)
725   FØRMAT(12X,I5,2E15.5)
      IF(MD.EQ.4) GØ TØ 726
      DØ 727 I=1,50
      EI=I
727   RN(I)=(EI-1.0)*DL
726    GØ TØ 260
C      DATA PREPARATIØNS FØR CØNSTANT WEIGHTING FUNCTIØN FØR INVERSE CHECK
730   DØ 740 I=1,N1
740   U(I)=1.0
      W(1)=1.0
      GØ TØ 620
C      DATA PREPARATIØNS FØR TRAPEZØID WEIGHTING FUNCTIØN FØR INVERSE CHECK
760   DØ 790 I=1,N1
      FI=I
      IF (FI.LT.S1) GØ TØ 770
      IF (FI.LT.S2) GØ TØ 780
      U(I)=0.0
      GØ TØ 790
770   U(I)=1.0
      GØ TØ 790
780   U(I)=(S2-FI)/(S2-S1)
790   CØNTINUE
      W(1)=U(1)
      GØ TØ 620
C      END ØF CALCULATIØNS
800   WRITE(IPR,810)
810    FØRMAT(//20X,10HEND ØF RUN)
      STØP
      END
C
C
C      TRANSFØRMATIØN TØ TRAPEZØID RULE APPRØXIMATIØN.CØNSTRUCTIØN ØF D MATRIX
       FUNCTIØN D(I,J)
       IF(J.GE.I+2) GØ TØ 2
      D=0.5
       RETURN
2      D=0.0
       RETURN
        END
C
C
       FUNCTIØN F(I,J)
      CØMMØN/BLØCK1/A,N,ISTART,NN,MD,MM
       IF(I.GE.N-1) GØ TØ 3
       IF(J.EQ.I) GØ TØ 1
       IF(MØD(J-I,2).EQ.0) GØ TØ 2
       IF(J.EQ.N) GØ TØ 5
      F=-2./3.*B(I,J)
       RETURN
```

```
1       F=7./3.*B(I,J)
         RETURN
2         IF(J.EQ.N-1) GO TO 5
         IF(J.EQ.N) GO TO 4
        F=8./3.*B(I,J)
         RETURN
3         F=.5*B(I,J)
        RETURN
4         F=1./3.*B(I,J)
         RETURN
5         F=2./3.*B(I,J)
         RETURN
         END
C
C
C       COMPUTES B MATRIX
        FUNCTION B (I,J)
C
C       ***********************************************************************
C
        COMMON/BLOCK1/A,N,ISTART,NN,MD,MM
        DATA PI/3.1415926/
         S=I
        T=J
        GO TO (10,20,20,30), MD
C        B MATRIX FOR CONSTANT WEIGHTING FUNCTION
10      B=PI*(2.0*T-A)/2.0
        RETURN
C        B MATRIX FOR GAUSSIAN AND DATA WEIGHTING FUNCTION
20      R=SQRT((T-A+1.0)**2-(S-1.0)**2)
        CALL WGHT23(XKI,TKS,R)
        B=(XKI+TKS)/(2.0-A)
        R=SQRT((T-1.0)**2-(S-1.0)**2)
        IF (J.EQ.I) RETURN
        CALL WGHT23(XKI,TKS,R)
        B=B-(XKI+TKS)/(2.0-A)
        RETURN
C        B MATRIX FOR TRAPEZOID WEIGHTING FUNCTION
30      R=SQRT((T-A+1.0)**2-(S-1.0)**2)
        CALL WGHT4(B1,R)
        B=(2.0*A-1.0)*B1
        R=SQRT((T-1.0)**2-(S-1.0)**2)
        IF (J.EQ.I) RETURN
        CALL WGHT4(B1,R)
        B=B-(2.0*A-1.0)*B1
        RETURN
        END
C
C       ***********************************************************************
        SUBROUTINE WGHT23(XKI,TKS,R)
C
C
C        CALCULATE B(I,J) FOR DATA WEIGHTING FUNCTION
C       ***********************************************************************
C
        DIMENSION TK(400)
        COMMON/BLOCK1/A,N,ISTART,NN,MD,MM
        COMMON/BLOCK2/RN(400),W(400),V(400),C(400),DL,G
        COMMON /BLOCK3/ KB,KL
        DATA PI/3.1415926/
```

```
      KC=KB+1
      DO 40 K=KC,KL
      IF (R.LT.RN(K)) GO TO 50
40    CONTINUE
      WF=W(KL)+(W(KL)-W(KL-1))*(R-RN(KL))/(RN(KL)-RN(KL-1))
      LA=KL
      GO TO 60
50    LA=K-1
      WF=W(LA)+(W(K)-W(LA))*(R-RN(LA))/(RN(K)-RN(LA))
60    XK0=R*R-RN(LA)**2
      IF (XK0.NE.0.) GO TO 70
      XKI=PI*R*R*0.250*(W(LA)+W(LA-1)-(RN(LA)+RN(LA-1))*(W(LA)-W(LA-1))/
     2(RN(LA)-RN(LA-1)))
      GO TO 80
70    XK1=SQRT(XK0)
      XK2=RN(LA)/XK1
      XK3=(R*R*ATAN(XK2)+RN(LA)*XK1)*RN(LA)+XK1**3/1.5
      XK4=(W(LA-1)-(DL/(R-RN(LA)))*(W(LA)-WF)-W(LA))*XK3/DL
      IF (LA.EQ.1) XK4=0.0
      XK5=PI*R*R*0.250*(WF+W(LA)-(R+RN(LA))/(R-RN(LA))*(WF-W(LA)))
      XKI=XK4+XK5
80    TKS=(W(2)-W(1))*R*R*R/(1.5*DL)
      IF (LA.EQ.1) TKS=(WF-W(1))*R*R*R/(1.5*DL)
      KK=LA-1
      IF (KK.LT.KC) RETURN
      DO 110 K=KC,KK
      TKD=R*R-RN(K)**2
      IF (TKD.NE.0.) GO TO 90
      TKA=PI*R*R*0.5*RN(K)/DL
      GO TO 100
90    TKA=RN(K)*R*R*ATAN(RN(K)/SQRT(TKD))+RN(K)*RN(K)*SQRT(TKD)+(SQRT(TK
     2D**3))/1.5
100   TK(K)=(W(K+1)+W(K-1)-2.0*W(K))*TKA/DL
      TKS=TKS+TK(K)
110   CONTINUE
      RETURN
      END
C
C     ******************************************************************
C
      SUBROUTINE WGHT4(B1,R)
C
C     ******************************************************************
      COMMON /BLOCK5/S1,S2
      DATA PI/3.1415926/
C      CALCULATES B(I,J) FOR TRAPEZOID WEIGHTING FUNCTION
C
      IF (R.LE.S1) GO TO 130
      IF (R.LE.S2) GO TO 120
      B2=S2*SQRT(R*R-S2*S2)
      B3=S1*SQRT(R*R-S1*S1)
      B6=S1*S1/B3
      B7=S2*S2/B2
      B4=R*R*ATAN(B7)
      B5=R*R*ATAN(B6)
      IF(S1.LT.0.000001) GO TO 115
      B1=(B3+B5)*0.5+(B2-B3+B4-B5)*0.5*S2/(S2-S1)+((B2/S2)**3-(B3/S1)**3
     2)/(3.0*(S2-S1))
      RETURN
115    B1=(B2+B4)*0.5+((B2/S2)**3-R**3)/(3.0*S2)
```

```
      RETURN
120   IF( S1.LT.0.000001 ) GØ TØ 140
      B3=S1*SQRT( R*R-S1*S1 )
      B6=S1*S1/B3
      B5=R*R*ATAN( B6 )
      B1=( B3+B5 )*0.5+( -B3-B5+R*R*PI/2.0 )*S2*0.5/( S2-S1 )-( ( B3/S1 )**3)/( 3.
     20*( S2-S1 ) )
      RETURN
130   B1=R*R*PI*0.25
      RETURN
140   B1=R*R*PI*0.25-( R**3 )/( 3.0*S2 )
      RETURN
      END
C
C
      SUBRØUTINE CGEN ( I )
C     ***********************************************************************
C      CALCULATE ELEMENTS ØF ITH RØW ØF MATRIX C
C
      CØMMØN/BLØCK1/A,N,ISTART,NN,MD,MM
      CØMMØN/BLØCK2/RN( 400 ),W( 400 ),V( 400 ),C( 400 ),DL,G
      DATA PI/3.1415926/
      S=I
      SN=N
C      CHECK FØR LØW ØRDER APPRØXIMATIØN
      IF ( MM.NE.0 ) GØ TØ 140
      IF ( I.EQ.N-1 ) GØ TØ 120
      IF ( I.EQ.N) GØ TØ 130
      DØ 110 J=I,N
      U=J
      IF ( I.EQ.1 ) GØ TØ 50
      IF ( I-J ) 10,30,40
10    IF ( MØD(J-I,2).EQ.0 ) GØ TØ 20
      IF ( J.EQ.N) GØ TØ 80
      U1=J
      U2=J-2
      U3=J-1
      F1=SQRT( U1*U1-( S-1.0 )**2 )
      F2=SQRT( U2*U2-( S-1.0 )**2 )
      T=( F1**3-F2**3 )/3.0+( S-1.0 )**2*( F1-F2 )-U3*( U1*F1-U2*F2 )-U3*( S-1.0 )
     2**2*ALØG( ( U1+F1 )/( U2+F2 ) )+U3*U3*( F1-F2 )
      V( J )=F1-F2-T
      GØ TØ 110
20    U1=J-1
      U2=J-3
      U3=J-2
      F1=SQRT( U1*U1-( S-1.0 )**2 )
      F2=SQRT( U2*U2-( S-1.0 )**2 )
      T=( F1**3-F2**3 )/3.0+( S-1.0 )**2*( F1-F2 )-U3*( U1*F1-U2*F2 )-U3*( S-1.0 )
     2**2*ALØG( ( U1+F1 )/( U2+F2 ) )+U3*U3*( F1-F2 )
      D=0.5*( U1*F1-U2*F2+( S-1.0 )**2*ALØG( ( U1+F1 )/( U2+F2 ) ) )-U3*( F1-F2 )
      V( J )=0.5*( D+T )
      IF ( J.EQ.N) GØ TØ 110
      IF ( J.EQ.N-1 ) GØ TØ 90
      U1=J+1
      U2=J-1
      U3=J
      F1=SQRT( U1*U1-( S-1.0 )**2 )
      F2=SQRT( U2*U2-( S-1.0 )**2 )
      T=( F1**3-F2**3 )/3.0+( S-1.0 )**2*( F1-F2 )-U3*( U1*F1-U2*F2 )-U3*( S-1.0 )
```

```
     2**2*ALOG((U1*F1)/(U2*F2))+U3*U3*(F1-F2)
      D=0.5*(U1*F1-U2*F2+(S-1.0)**2*ALOG((U1*F1)/(U2*F2)))-U3*(F1-F2)
      V(J)=V(J)-0.5*(D-T)
      GO TO 110
30    U1=I+1
      U2=I-1
      U3=I
      F1=SQRT(4.0*S)
      T=F1**3/3.0+(S-1.0)**2*F1-U3*U1*F1-U3*(S-1.0)**2*ALOG((U1+F1)/U2)
     1+U3*U3*F1
      D=0.5*(U1*F1+(S-1.0)**2*ALOG((U1+F1)/U2))-U3*F1
      V(J)=-0.5*(D-T)
      GO TO 110
40    V(J)=0.0
      GO TO 110
50    IF (I-J) 60,70,100
60    V(J)=1.0
      GO TO 110
70    V(1)=0.5
      GO TO 110
80    F1=SQRT((U-1.0)**2-(S-1.0)**2)
      F2=SQRT((U-2.0)**2-(S-1.0)**2)
      V(J)=(F1-F2)*0.5
      GO TO 110
90    F1=SQRT(U*U-(S-1.0)**2)
      F2=SQRT((U-1.0)**2-(S-1.0)**2)
      V(J)=V(J)+(F1-F2)*0.5
      GO TO 110
100   V(J)=0.0
110   CONTINUE
      RETURN
C      END POINTS CALCULATIONS
120   F4=SQRT(2.0*(SN-1.0)-1.0)
      V(N-1)=0.5*F4
      V(N)=0.5*F4
      RETURN
C      LOW-ORDER APPROXIMATION
130   V(N)=0.5*SQRT(2.0*SN-1.0)
      RETURN
140   DO 190 J=I,N
      U=J
      IF (I-J) 150,160,170
150   IF (J.EQ.N) GO TO 180
      V(J)=(SQRT(U*U-(S-1.0)**2)-SQRT((U-2.0)**2-(S-1.0)**2))*0.5
      GO TO 190
160   V(J)=(SQRT(2.0*S-1.0))*0.5
      GO TO 190
170   V(J)=0.0
      GO TO 190
180   V(J)=0.5*(SQRT((U-1.0)**2-(S-1.0)**2)-SQRT((U-2.0)**2-(S-1.0)**2))
190   CONTINUE
      RETURN
      END
C
C     CORRECTION FOR THE EFFECTS OF THE WIDTH USING THE METHOD OF TAYLOR
C     AND SCHMIDT
      SUBROUTINE WIDTH(FF,LW,WQ,KO)
      DIMENSION FF(400),F(400),LW(10),WQ(10)
      COMMON/BLOCK1/A,N,ISTART,NN,MD,MM
      AA=756.
```

```
      LW(KØ+1)=N+1
      DØ 1 K=1,KØ
      NW=LW(K)+2
      KW=LW(K+1)-4
      DØ 305 J= NW,KW
  305 F(J) = FF(J)-WQ* (FF(J-2)*120.-FF(J-1)*87.-FF(J)*96.-FF(J+1)*12.+
     1 FF(J+2)*60.+FF(J+3)*15.) / AA
      J0=LW(K)
      IMAX=LW(K+1)-1
      F(J0) = FF(J0)-WQ*(330.*FF(J0)-381.*FF(J0+1)-264.*FF(J0+2)+156.*FF
     1(J0+3)+354.*FF(J0+4)-195.*FF(J0+5))/AA
      F(J0+1) = FF(J0+1)-WQ*(225.*FF(J0)-234.*FF(J0+1)-180.*FF(J0+2)+72.
     1*FF(J0+3)+207.*FF(J0+4)-90.*FF(J0+5))/AA
      F(IMAX-2) = FF (IMAX-2) - WQ*( 15.*FF(IMAX-5)+ 60.*FF(IMAX-4)-12.*
     1FF(IMAX-3)- 96.*FF(IMAX-2)- 87.*FF(IMAX-1)+120.*FF(IMAX)) /AA
      F(IMAX-1) = FF (IMAX-1) - WQ*(-90.*FF(IMAX-5)+207.*FF(IMAX-4)+72.*
     1FF(IMAX-3)-180.*FF(IMAX-2)-234.*FF(IMAX-1)+225.*FF(IMAX)) /AA
      F(IMAX)  = FF(IMAX)  -WQ*(-195.*FF(IMAX-5) +354.*FF(IMAX-4) +156.*
     1FF(IMAX-3) -264.*FF(IMAX-2) -381.*FF(IMAX-1)+330.*FF(IMAX)) /AA
1     CØNTINUE
      DØ 2 J=1,N
2     FF(J)=F(J)
      RETURN
      END
```

CHAPTER 6

1. TYPICAL CALCULATIONS AND DISCUSSION OF VARIOUS ALGORITHMS

In Figures 1-4, unsmeared data calculated from the measurements supplied by J. J. Weeks of this laboratory are presented. The measurements were performed on melt-crystallized polyethylene.

In Figure 1, results based on very low scattering angles are shown. Results based on Nÿstrom algorithm (forward interpolation) are shown by a solid line. Results based on the rectangle rule approximation and on the trapezoid rule approximation (uncorrected for perturbation effects) are shown by sets of points. As is evident, even-odd oscillations start to be noticeable at larger angles.

In Figure 2, results for larger angles are shown ($2\theta > .008$ rad.). The bottom set of points represent the input data. The desmeared data are shown for two algorithms: rectangle rule approximation and Nÿstrom algorithm. While both methods of interpolations fall on practically the same curve, the Nÿstrom algorithm seems to lead to more numerically stable results (smaller oscillations from the smooth curve).

As we pointed above, the results for desmearing data are very sensitive to the numerical perturbations in the input data. If the data show excessive fluctuations (resulting from experimental error or random noise effects), those will be magnified when desmeared. To demonstrate this, we present, in Figure 3, results obtained from experimental data with certain amounts of random noise (bottom curve in Figure 3). The upper curve and points show results of desmearing data represented by the bottom curve. The sharp fluctuations in the desmeared data are present, no matter what algorithm is used in the calculations. In this

figure, the solid line represents the results based on the Nÿstrom algorithm. These results are somehow more smooth than the ones based on other algorithms. On the other hand, the trapezoid rule approximation, even after correction based on perturbation method (smoothing the results) were applied, lead to the biggest scatter in the calculated data.

In Figures 1-3 a constant weighting function was assumed. In Figure 4, trapezoid weighting function was assumed. As expected, the presence of a weighting function has an effect on stabilizing the numerical solution. The unsmeared data are devoid of scatter, and all three interpolating methods employed (see captions to Figure 5 for their description) fall into identical curve.

In Figure 5 we demonstrate the effect of desmearing on peak formation in the scattering curve. The original (input) data were provided by D. McIntyre of the Polymer Institute, University of Akron. The measurements were performed on a triblock polymer of polystyrene-polybutadiene-polystyrene.

The lower curve in Figure 5 represents the original data. The upper curve shows the desmeared data, obtained with the Nÿstrom algorithm (forward interpolation). Of interest is the existence of the first peak in the desmeared curve (indicated by an arrow); there is no corresponding maximum in the input data, only an inflection point. We also notice the following facts: 1) the maximum in the input data are greatly magnified when these data are desmeared; and 2) the desmearing of the scattering curve tends to shift the locations of the peaks toward slightly larger angles.

In all these Figures, only a part of the scattering curves are shown. In Table 3 and 4 typical calculations are presented, together with the input data. Fragments of the input and output data presented in Table 3 and 4 are depicted in Figures 4 and 5.

2. TEST FUNCTIONS

Two functions were employed using a constant weighting function:

1) I(x) is Gaussian. This function was employed in order to compare the accuracy of the various algorithms employed and in order to check the effect of the cut-off on the relative error. The function employed is:

$$\tilde{I}_i = e^{-i^2/1250} \quad \text{for } i=0, 1, \ldots, 100. \tag{49}$$

Tabulated values of $I_i/\tilde{I}_i$, using various algorithms, are shown in Table 5. The exact results are computed from:

$$\frac{I_i}{\tilde{I}_i} = 0.0159575 \left\{ 1 + \frac{\exp[-(10{,}000-i^2)/1250]}{2\left(\frac{10{,}000-i^2}{1250}\right)^{3/2} \pi^{1/2}} \right\}. \tag{50}$$

As is evident, Nÿstrom algorithm leads to best agreement with exact results. At x=90, the error because of the cut-off at x=100, is still about 1%.

2) The function was taken from P. Schmidt [10]. It represents the infinite-slit smeared scattering function for monodisperse spheres. This function corresponds to the perfect collimation intensity function with a rapid succession of maxima of diminishing intensities. Hence, it provides a good test for the numerical method employed.

The perfect collimation intensity I(x) for spheres is given by

$$I_i = 9\,[\sin(ia) - ia\cos(ia)]^2/_{(ia)^6} \tag{51}$$

i is a scattering angle. a=0.25; i=3,4, ..., 80. The results are shown in Figure 6. In this figure, ia=1 corresponds to a scattering angle of 4 miliradians. The solid line represents eq. (51). The points are the calculated intensities. Method of forward interpolation (Nÿstrom algorithm) was employed in the calculations.

As is evident from Figure 6, there is a close agreement between the calculated and exact results, almost up to i=76. The cut-off of calculations was at i=80 (corresponding to scattering angle of 80 miliradians).

The author wishes to thank Dr. John Smith for initiating the computer program and to Mr. James Weeks for helpful suggestions about handling experimental information and also for providing experimental data. Mr. Martin Cordes' help in preparation of this Technical Note, and in particular, his help with the analysis of the mathematical background to this work, is very appreciated.

REFERENCES

[1] Taylor, T. R., and Schmidt, P. W., Acta Physica Austriaca, 25, 293 (1967).

[2] Hendricks, R., and Schmidt, P. W., Acta Physica Austriaca, 26, 97 (1967).

[3] Guinier, A., Fournet, G., Walker, C. B., and Yudowitch, K., Small Angle Scattering of x-rays, p. 114 (John Wiley and Sons, Inc., N.Y., 1955).

[4] Shull, C. G., and Roess, L. C., J. Appl. Phys., 18, 295 (1947).
Krathky, O., Porod, G., and Skala, Z., Acta Physica Austriaca, 13, 76 (1960).
Schmidt, P. W., Acta Crystallagraphica, 19, 938 (1965).
Gerold, V., Acta Crystallorgraphica, 10, 287 (1957).

[5] Vonk, G. C., J. Appl. Crystallographica, 4, 340 (1971). In this article a different algebraic approach for the slit-height desmearing is presented, based on the original eq. (2), rather than on the transformed eq. (3).

[6] Mazur, J., J. Research Natl. Bur. Stand. (U.S.), 75B, No. 3 & 4, 173 (1971).
Mazur, J., and Wims, A., J. Res. Natl. Bur. Stand. (U.S.), 70A, No. 6, 467 (1966).

[7] Delves, L. M., and Walsh, J., Numerical Solution of Integral Equations, Clarendon Press, Oxford, Chapter 12 (1974).
Schmacdeke, W. W., Approximate Solutions for Volterra Integral Equations of the First Kind, Journal of Mathematical Analysis and Applications, 23, pp. 604-613 (1968).

[8] Kobayasi, M., Rep. Stat. Appl. Res. J.U.S.E., 14, 65 (1967).

[9] Linz, P., The Numerical Solution of Volterra Integral Equations by Finite Difference Methods, MRC Tech. Summary Rep. 825, Mathematics Research Center, Univ. of Wisconsin, Madison (1967).

[10] Schmidt, P. W., Acta Crystallographica, I, 722 (1955).

[11] Fox, L., An Introduction of Linear Matrix Algebra, Chapter 2, Oxford University Press (1965).

[12] Henrici, D., Discrete Variable Methods is Ordinary Differential Equations (John Wiley and Sons, Inc., N.Y., 1962).

[13] Hildebrand, R., Finite Difference Equations and Simulations, Chapter 2, (Prentice-Hall, Inc., Englewood Cliffs, N.Y., 1968).

TABLE 1

SYMBOL	FORMAT	CONDITION	COMMENT
NN ISTART IENTR NSUB	4I10	First card.	IENTR=0 indicates that there is no interpolation of input data.
LW* WM2*	I10, F10.5 (per card)	Number of cards = number of slit-width corrections + 1 (blank)	Last card must be blank. If no slit-width correction applies a single blank card must be inserted.
MD IDC INV MM	4I10	Single card	
G	F10.2	MD = 2**	
RN* W*	2F10.2	MD = 3**	Series of cards. Last card must be blank.
S1 S2	2F10.2	MD = 4**	
XI*	8F10.0 (per card)		Last card must be blank.

*Dimensioned variables.

**These cards are inserted only according to the directive given by the value of the integer MD.

Table 2

ARRAYS USED IN PROGRAM

SYMBOL	DESCRIPTION
XI	Input Vector
YY	Output Vector, rectangular rule approximation
YI	" " , trapezoid rule approximation
YK	" " , smoothing correction applied to YI
YL	" " , smoothing correction applied to YK
YM	" " , Nÿstrom algorithm (forward interpolation)
XX	Temporary input vector, for data interpolation.
ZI	Input vector, recalculated by solution checking (used only if INV≠0)
W	Input weighting function
RN	Input coordinates of W
U	Temporary storage for weighting function for solution checking (used only if INV≠0)
V	Array of γ_{ij} for given i
C	$\sum_{j=i}^{N} V_{ij} \tilde{X}_j$

Table 5

Exact and Computed Ratios of Intensities $I_i/\tilde{I}_i$ for Various Algorithms

$\tilde{I}_i = \exp -(i^2/1250), \quad i<100$

i	Exact (Eq. 50)	Rectangle	Trapezoid	Smoothed Trapezoid	Nÿstrom
1	.0159575	.0159386	.0159557	.0119487	.0159699
10	.0159575	.0158244	.0159576	.0159503	.0159586
20	.0159575	.0156990	.0159560	.0159533	.0159576
30	.0159575	.0155747	.0159538	.0159584	.0159573
40	.0159575	.0154519	.0159503	.0159655	.0159573
50	.0159582	.0153306	.0159447	.0159748	.0159579
60	.0159598	.0152115	.0159340	.0159863	.0159594
70	.0159668	.015098	.0159054	.0160107	.0159645
80	.0160092	.015010	.015813	.016033	.015988
90	.0163957	.015171	.015541	.0162521	.0161731

TABLE 3

SAMPLE DATA DECK

```
100

   4          1          1
  25.00      60.00
37.40000   13.80000   10.30000   8.68000   7.65000   6.67000   5.81000   4.89000
 4.00000    3.20000    2.40000   1.81000   1.32000   0.99000   0.74000   0.59500
 0.52000    0.46500    0.42000   0.39000   0.35900   0.34100   0.31400   0.27300
 0.25500    0.22000    0.20100   0.17700   0.16000   0.14000   0.12400   0.11000
 0.10100    0.09500    0.08900   0.08200   0.07750   0.07350   0.07000   0.06650
 0.06300    0.06050    0.05800   0.05650   0.05450   0.05200   0.04950   0.04700
 0.04500    0.04300    0.04150   0.04000   0.03840   0.03700   0.03520   0.03400
 0.03290    0.03180    0.03100   0.02990   0.02910   0.02860   0.02800   0.02760
 0.02720    0.02690    0.02640   0.02580   0.02540   0.02490   0.02430   0.02380
 0.02320    0.02280    0.02230   0.02180   0.02120   0.02070   0.02030   0.01990
 0.01950    0.01920    0.01900   0.01880   0.01850   0.01830   0.01810   0.01790
 0.01770    0.01750    0.01740   0.01720   0.01710   0.01700   0.01690   0.01680
 0.01670    0.01660    0.01650   0.01640
```

SPECIFICATIONS
N,TOTAL RANGE OF DATA,=100 NN,NUMBER OF DATA READ=100 ISTART,STARTING POINT,= 0

W IS TRAPEZOID,S1= 25.000,S2= 60.000

C-VALUES..

0	.98402+02	42	.29645+01	84	.92378+00
1	.83610+02	43	.28816+01	85	.88917+00
2	.76681+02	44	.27949+01	86	.85343+00
3	.70560+02	45	.27083+01	87	.81716+00
4	.64307+02	46	.26247+01	88	.77947+00
5	.57646+02	47	.25463+01	89	.74145+00
6	.50816+02	48	.24733+01	90	.70166+00
7	.43717+02	49	.24040+01	91	.66001+00
8	.37055+02	50	.23397+01	92	.61660+00
9	.30766+02	51	.22752+01	93	.57013+00
10	.25355+02	52	.22141+01	94	.51976+00
11	.21014+02	53	.21534+01	95	.46425+00
12	.17652+02	54	.20961+01	96	.40143+00
13	.15193+02	55	.20444+01	97	.32734+00
14	.13003+02	56	.19944+01	98	.23089+00
15	.12239+02	57	.19471+01	99	.11569+00
16	.11369+02	58	.19016+01		
17	.10643+02	59	.18567+01		
18	.99876+01	60	.18166+01		
19	.94178+01	61	.17774+01		
20	.88469+01	62	.17394+01		
21	.83429+01	63	.17019+01		
22	.77235+01	64	.16647+01		
23	.72106+01	65	.16265+01		
24	.67074+01	66	.15867+01		
25	.62452+01	67	.15471+01		
26	.58335+01	68	.15086+01		
27	.54593+01	69	.14685+01		
28	.51182+01	70	.14292+01		
29	.48080+01	71	.13957+01		
30	.45425+01	72	.13514+01		
31	.43229+01	73	.13136+01		
32	.41454+01	74	.12755+01		
33	.39885+01	75	.12373+01		
34	.38337+01	76	.11995+01		
35	.36922+01	77	.11635+01		
36	.35697+01	78	.11284+01		
37	.34539+01	79	.10934+01		
38	.33454+01	80	.10592+01		
39	.32309+01	81	.10259+01		
40	.31417+01	82	.99285+00		
41	.30501+01	83	.95867+00		

ARBITRARY UNITS
INPUT DATA EXTRAPOLATED TO ZERO ANGLE

I	INPUT VECTOR YI(I)	ZEROTH ORDER YY(I)	TRAPEZOID YT(I)	TRAPEZOID+PERT. YK(I)	TRAPEZOID+2PERT. YL(I)	NYSTROM INTER. YM(I)
0	.374000+02	.941799+01	.168500+02	.168500+02	.168500+02	.000000
1	.168000+02	.147069+01	.198602+01	.429164+01	.584756+01	.326784+01
2	.103000+02	.779972+00	.955756+00	.106887+01	.153638+01	.102463+01
3	.868000+01	.569320+00	.604189+00	.651160+00	.699782+00	.656559+00
4	.765000+01	.471755+00	.534451+00	.525176+00	.529571+00	.515821+00
5	.667000+01	.395878+00	.409059+00	.425564+00	.433837+00	.433296+00
6	.581000+01	.348199+00	.382697+00	.375591+00	.373827+00	.372754+00
7	.489000+01	.283215+00	.313699+00	.315037+00	.315518+00	.315484+00
8	.400000+01	.236164+00	.252731+00	.257370+00	.259510+00	.260375+00
9	.320000+01	.181866+00	.219598+00	.212543+00	.210040+00	.207721+00
10	.249000+01	.132104+00	.144134+00	.152701+00	.156433+00	.155251+00
11	.181000+01	.935789-01	.120075+00	.115253+00	.114719+00	.111273+00
12	.132000+01	.631338-01	.670825-01	.745984-01	.781095-01	.770785-01
13	.990000+00	.427127-01	.591851-01	.550106-01	.543833-01	.516637-01
14	.740000+00	.260245-01	.262403-01	.316592-01	.341725-01	.331945-01
15	.595000+00	.183302-01	.258087-01	.233878-01	.232936-01	.216987-01
16	.520000+00	.144720-01	.108516-01	.145513-01	.160945-01	.162414-01
17	.465000+00	.123578-01	.180924-01	.149741-01	.140311-01	.132851-01
18	.420000+00	.102354-01	.662317-02	.973820-02	.108674-01	.111945-01
19	.390000+00	.974181-02	.138477-01	.112750-01	.104518-01	.985671-02
20	.359000+00	.823824-02	.563590-02	.787198-02	.871470-02	.904751-02
21	.341000+00	.949694-02	.108406-01	.952526-02	.907150-02	.899889-02
22	.314000+00	.771770-02	.815329-02	.945598-02	.845811-02	.854245-02
23	.273000+00	.719869-02	.728211-02	.739950-02	.749181-02	.748865-02
24	.255000+00	.637677-02	.711527-02	.689691-02	.680443-02	.677101-02
25	.220000+00	.550336-02	.563826-02	.583946-02	.591585-02	.591250-02
26	.201000+00	.464911-02	.536945-02	.524031-02	.521537-02	.516678-02
27	.177000+00	.429566-02	.432976-02	.449151-02	.455437-02	.458257-02
28	.160000+00	.380249-02	.426156-02	.411990-02	.406970-02	.405975-02
29	.140000+00	.319542-02	.334341-02	.344711-02	.347680-02	.348836-02
30	.124000+00	.261458-02	.304744-02	.295248-02	.292646-02	.287215-02
31	.110000+00	.210894-02	.215172-02	.230175-02	.235937-02	.232659-02
32	.101000+00	.184520-02	.203617-02	.199677-02	.200765-02	.197443-02
33	.950000-01	.177183-02	.165423-02	.175709-02	.179647-02	.181919-02
34	.890000-01	.159973-02	.158943-02	.175366-02	.170054-02	.167821-02
35	.820000-01	.138472-02	.131002-02	.143149-02	.147628-02	.148201-02
36	.775000-01	.129035-02	.145941-02	.137806-02	.135751-02	.133433-02
37	.735000-01	.119334-02	.112968-02	.120136-02	.123090-02	.124321-02
38	.700000-01	.114633-02	.126600-02	.120199-02	.118016-02	.117199-02
39	.665000-01	.104302-02	.102666-02	.107202-02	.108583-02	.109211-02
40	.630000-01	.972527-03	.105949-02	.102503-02	.101549-02	.100420-02
41	.605000-01	.904241-03	.865560-03	.920776-03	.934114-03	.931836-03
42	.580000-01	.862474-03	.922821-03	.896546-03	.892015-03	.882722-03
43	.565000-01	.870865-03	.862034-03	.845126-03	.860664-03	.871119-03
44	.545000-01	.850242-03	.939697-03	.886935-03	.866692-03	.863706-03
45	.520000-01	.810829-03	.760788-03	.807286-03	.820963-03	.831494-03
46	.495000-01	.757400-03	.860871-03	.809700-03	.791928-03	.783344-03
47	.470000-01	.705104-03	.653930-03	.705478-03	.723491-03	.730357-03
48	.450000-01	.666619-03	.756278-03	.709334-03	.694042-03	.683863-03
49	.430000-01	.620734-03	.576961-03	.621438-03	.637783-03	.642580-03

PLOTS

50	.415000-01	.609593-03	.664507-03	.631612-03	.621426-03	.615614-03
51	.400000-01	.577740-03	.554679-03	.580671-03	.589174-03	.594325-03
52	.384000-01	.563704-03	.600800-03	.580743-03	.574135-03	.572989-03
53	.370000-01	.533132-03	.526609-03	.541148-03	.544431-03	.547322-03
54	.352000-01	.490953-03	.539654-03	.521246-03	.515779-03	.510865-03
55	.340000-01	.472415-03	.442252-03	.468540-03	.478327-03	.481280-03
56	.329000-01	.448669-03	.502579-03	.474554-03	.465284-03	.460255-03
57	.318000-01	.431413-03	.394758-03	.424946-03	.435754-03	.441402-03
58	.310000-01	.421069-03	.468067-03	.440183-03	.429970-03	.426153-03
59	.299000-01	.387283-03	.374071-03	.394141-03	.400785-03	.402600-03
60	.291000-01	.376110-03	.400496-03	.387963-03	.384961-03	.381785-03
61	.286000-01	.363479-03	.351723-03	.363770-03	.367908-03	.369300-03
62	.280000-01	.354498-03	.375234-03	.364404-03	.361029-03	.358317-03
63	.276000-01	.346108-03	.333761-03	.344789-03	.348922-03	.349680-03
64	.272000-01	.345351-03	.358455-03	.349971-03	.347699-03	.346750-03
65	.269000-01	.346963-03	.332246-03	.341520-03	.344248-03	.346888-03
66	.264000-01	.339672-03	.361680-03	.349438-03	.344647-03	.342819-03
67	.258000-01	.328494-03	.317664-03	.328610-03	.332783-03	.334101-03
68	.254000-01	.330633-03	.339323-03	.332817-03	.330728-03	.330437-03
69	.249000-01	.320897-03	.321944-03	.324492-03	.324945-03	.325892-03
70	.243000-01	.315907-03	.319851-03	.318885-03	.318525-03	.318766-03
71	.238000-01	.305484-03	.311964-03	.311118-03	.310696-03	.310042-03
72	.232000-01	.297971-03	.299005-03	.300820-03	.301803-03	.301822-03
73	.228000-01	.294354-03	.296937-03	.296421-03	.296437-03	.296411-03
74	.223000-01	.290231-03	.291771-03	.292119-03	.292062-03	.293161-03
75	.219000-01	.283498-03	.288691-03	.287473-03	.286657-03	.286917-03
76	.212000-01	.271985-03	.278305-03	.277929-03	.277754-03	.278203-03
77	.207000-01	.264347-03	.265664-03	.267332-03	.268239-03	.268631-03
78	.203000-01	.259156-03	.263030-03	.262177-03	.261903-03	.264378-03
79	.199000-01	.251700-03	.255292-03	.255379-03	.255137-03	.254322-03
80	.195000-01	.241561-03	.248118-03	.247126-03	.246824-03	.245810-03
81	.192000-01	.234621-03	.235003-03	.237062-03	.238189-03	.237957-03
82	.190000-01	.232429-03	.254239-03	.233763-03	.233780-03	.233632-03
83	.188000-01	.228652-03	.230619-03	.230566-03	.230254-03	.230467-03
84	.185000-01	.221144-03	.226685-03	.225494-03	.225178-03	.224443-03
85	.183000-01	.218829-03	.215604-03	.218526-03	.219658-03	.220067-03
86	.181000-01	.214170-03	.222054-03	.216351-03	.217154-03	.216367-03
87	.179000-01	.212531-03	.206286-03	.210995-03	.212501-03	.213285-03
88	.177000-01	.206709-03	.218777-03	.212672-03	.210677-03	.209057-03
89	.175000-01	.205774-03	.194641-03	.202375-03	.205210-03	.206284-03
90	.174000-01	.204593-03	.216907-03	.209090-03	.206415-03	.204789-03
91	.172000-01	.202395-03	.192273-03	.199753-03	.202387-03	.203960-03
92	.171000-01	.203755-03	.212517-03	.206222-03	.204387-03	.204361-03
93	.170000-01	.206289-03	.194593-03	.201679-03	.204083-03	.205349-03
94	.169000-01	.210806-03	.217586-03	.211560-03	.209984-03	.200818-03
95	.168000-01	.221114-03	.204026-03	.211982-03	.215804-03	.192538-03
96	.167000-01	.246700-03	.238202-03	.235338-03	.237600-03	.186856-03
97	.166000-01	.314886-03	.255199-03	.272262-03	.279718-03	.343493-03
98	.165000-01	.372314-03	.374573-03	.353924-03	.281326-03	.723343-03
99	.164000-01	.370056-03	.370056-03	.000000	.000000	.111279-02

BACK CHECKS
INPUT RECOMPUTED

I	XI(I)	ZI(I)			
0	.37400+02	.24612+02	50	.41500-01	.40985-01
1	.13800+02	.13395+02	51	.40000-01	.39121-01
2	.10300+02	.10145+02	52	.38400-01	.37905-01
3	.86800+01	.85949+01	53	.37000-01	.36136-01
4	.76500+01	.75566+01	54	.35200-01	.34649-01
5	.66700+01	.66062+01	55	.34000-01	.33254-01
6	.58100+01	.58016+01	56	.32900-01	.32338-01
7	.48900+01	.48465+01	57	.31800-01	.31097-01
8	.40000+01	.39953+01	58	.31000-01	.30489-01
9	.32000+01	.31603+01	59	.29900-01	.29151-01
10	.24000+01	.23713+01	60	.29100-01	.28603-01
11	.18100+01	.18002+01	61	.28600-01	.27922-01
12	.13200+01	.13040+01	62	.28000-01	.27502-01
13	.99000+00	.98528+00	63	.27600-01	.26951-01
14	.74000+00	.71889+00	64	.27200-01	.26724-01
15	.59500+00	.59830+00	65	.26900-01	.26321-01
16	.52000+00	.50683+00	66	.26400-01	.25902-01
17	.46500+00	.46964+00	67	.25800-01	.25224-01
18	.42000+00	.40958+00	68	.25400-01	.24954-01
19	.39000+00	.39539+00	69	.24900-01	.24336-01
20	.35900+00	.34873+00	70	.24300-01	.23853-01
21	.34100+00	.34648+00	71	.23600-01	.23269-01
22	.31400+00	.30369+00	72	.23200-01	.22744-01
23	.27300+00	.27738+00	73	.22800-01	.22318-01
24	.25500+00	.24937+00	74	.22300-01	.21849-01
25	.22000+00	.22063+00	75	.21800-01	.21336-01
26	.20100+00	.19793+00	76	.21200-01	.20730-01
27	.17700+00	.17623+00	77	.20700-01	.20233-01
28	.16000+00	.15802+00	78	.20300-01	.19836-01
29	.14000+00	.13829+00	79	.19900-01	.19407-01
30	.12400+00	.12246+00	80	.19500-01	.19000-01
31	.11000+00	.10826+00	81	.19200-01	.18691-01
32	.10100+00	.99984-01	82	.19000-01	.18521-01
33	.95000-01	.93872-01	83	.18800-01	.18377-01
34	.89000-01	.88021-01	84	.18500-01	.18155-01
35	.82000-01	.80444-01	85	.18300-01	.18125-01
36	.77500-01	.76642-01	86	.18100-01	.17939-01
37	.73500-01	.72082-01	87	.17900-01	.17672-01
38	.70000-01	.69315-01	88	.17700-01	.17511-01
39	.66500-01	.65177-01	89	.17500-01	.17711-01
40	.63000-01	.62344-01			
41	.60500-01	.59526-01	90	.17400-01	.17776-01
			91	.17200-01	.17807-01
42	.58000-01	.57343-01	92	.17100-01	.17837-01
43	.56500-01	.55509-01	93	.17000-01	.18133-01
44	.54500-01	.53833-01	94	.16900-01	.18216-01
45	.52000-01	.51002-01	95	.16800-01	.18418-01
46	.49500-01	.48799-01	96	.16700-01	.18953-01
47	.47000-01	.46074-01	97	.16600-01	.19310-01
48	.45000-01	.44404-01	98	.16500-01	.19162-01
49	.43000-01	.42071-01	99	.16400-01	.13487-01

@FIN END OF RUN NIFØ

TABLE 4

SAMPLE DATA DECK

```
128         7

  1                   1
6076       5777       5577       5361       5222       5109       4930       4826
4698       4594       4490       4384       4288       4188       4148       4114
4053       3979       3807       3468       2947       2221       1367        791
 583        558        577        609        628        639        630        614
 589        551        500        433        368        305        252        207
 176        158        150        148        152        155        158        160
 165        163        160        159        154        150        142        134
 122        109       94.9       84.9       75.8       68.2       63.5       59.8
58.1       58.7       59.5       59.9       61.2       62.2       62.0       63.7
60.9       61.7       59.0       56.9       55.0       53.5       51.2       47.5
45.5       42.3       39.4       36.6       34.6       32.0       30.8       29.8
29.5       28.6       28.8       28.8       28.6       28.5       28.9       29.0
29.3       29.3       29.3       29.4       28.7       28.1       27.8       26.1
24.9       22.4       20.7       19.2       17.9       17.0       16.2       15.8
15.8       15.8       16.2       16.4       16.4       15.9       16.0       15.4
14.4       13.9       12.8       12.2       10.9       10.3       9.76       9.51
```

SPECIFICATIONS

N,TOTAL RANGE OF DATA,=135 NN,NUMBER OF DATA READ=128 ISTART,STARTING POINT,= 7

W IS CONSTANT

ARBITRARY UNITS

I	INPUT VECTOR XI(I)	ZEROTH ORDER YY(I)	TRAPEZOID YT(I)	TRAPEZOID+PERT. YK(I)	TRAPEZOID+2PERT. YL(I)	NYSTROM INTER. YM(I)
7	.607600+04	.141759+03	.130052+03	.137279+03	.116902+03	.000000
8	.577700+04	.121642+03	.103465+03	.122290+03	.125203+03	.131128+03
9	.557700+04	.112383+03	.139819+03	.124782+03	.119988+03	.116242+03
10	.536100+04	.967098+02	.859475+02	.985136+02	.103257+03	.104067+03
11	.522200+04	.979292+02	.107472+03	.100704+03	.989904+02	.979185+02
12	.510900+04	.915276+02	.883963+02	.926144+02	.936497+02	.944204+02
13	.493000+04	.860158+02	.946588+02	.907374+02	.896208+02	.895583+02
14	.482600+04	.831039+02	.773628+02	.821608+02	.839934+02	.844316+02
15	.469800+04	.799909+02	.868451+02	.839800+02	.824256+02	.814134+02
16	.459400+04	.782963+02	.711368+02	.764727+02	.783383+02	.791725+02
17	.449000+04	.767437+02	.854438+02	.801592+02	.784069+02	.776304+02
18	.433400+04	.752085+02	.630436+02	.733319+02	.750852+02	.759479+02
19	.428800+04	.734904+02	.823734+02	.770241+02	.751995+02	.737116+02
20	.416800+04	.712085+02	.646074+02	.697688+02	.718654+02	.717361+02
21	.414800+04	.762621+02	.778096+02	.750934+02	.745773+02	.732888+02
22	.411800+04	.809877+02	.747146+02	.773215+02	.783779+02	.778899+02
23	.405300+04	.894163+02	.872607+02	.858882+02	.857960+02	.848668+02
24	.397900+04	.100717+03	.915719+02	.939018+02	.948221+02	.955999+02
25	.380700+04	.111732+03	.109362+03	.107437+03	.106197+03	.107830+03
26	.346800+04	.115263+03	.113601+03	.113532+03	.112692+03	.116818+03
27	.294700+04	.111571+03	.116926+03	.114587+03	.112172+03	.116829+03
28	.229100+04	.856671+02	.106215+03	.101151+03	.973818+02	.985268+02
29	.136700+04	.445148+02	.651186+02	.651002+02	.649115+02	.622325+02
30	.791000+03	.153256+02	.239110+02	.279171+02	.309654+02	.275775+02
31	.583000+03	.500539+01	.674628+01	.902374+01	.113056+02	.921071+01
32	.558000+03	.318811+01	.327031+01	.382124+01	.463389+01	.380135+01
33	.577000+03	.418990+01	.316590+01	.349464+01	.381836+01	.363208+01
34	.609000+03	.586739+01	.527389+01	.511039+01	.509126+01	.506047+01
35	.628000+03	.750582+01	.646089+01	.661137+01	.664931+01	.677217+01
36	.639000+03	.896337+01	.855675+01	.833998+01	.821320+01	.827834+01
37	.630000+03	.958440+01	.937600+01	.930792+01	.925041+01	.928386+01
38	.614000+03	.104151+02	.979279+01	.993075+01	.996581+01	.100453+02
39	.589000+03	.108394+02	.110374+02	.107640+02	.106464+02	.107348+02
40	.551000+03	.111942+02	.106415+02	.108917+02	.109171+02	.111611+02
41	.500000+03	.105743+02	.117668+02	.111718+02	.108898+02	.109684+02
42	.433000+03	.930339+01	.940185+01	.975936+01	.982584+01	.997300+01
43	.368000+03	.772316+01	.920492+01	.874302+01	.856978+01	.852071+01
44	.305000+03	.600601+01	.624140+01	.668352+01	.683034+01	.689561+01
45	.252000+03	.449584+01	.593062+01	.550415+01	.536894+01	.524741+01
46	.207000+03	.298367+01	.306106+01	.351353+01	.369002+01	.366897+01

47	.176000+03	.185556+01	.290628+01	.258184+01	.251962+01	.235690+01
48	.158000+03	.117014+01	.804843+00	.127685+01	.147460+01	.145925+01
49	.150000+03	.769560+00	.153543+01	.115838+01	.106458+01	.924647+00
50	.148000+03	.658164+00	.368735-02	.477137+00	.665146+00	.699621+00
51	.152000+03	.801045+00	.131264+01	.923950+00	.806459+00	.732213+00
52	.155000+03	.906959+00	.289448+00	.665817+00	.738759+00	.842497+00
53	.158000+03	.100459+01	.152447+01	.114534+01	.102107+01	.937598+00
54	.160000+03	.114847+01	.494639+00	.879247+00	.102744+01	.109859+01
55	.165000+03	.154620+01	.181225+01	.150231+01	.138592+01	.136143+01
56	.163000+03	.145475+01	.128015+01	.142703+01	.146365+01	.149111+01
57	.160000+03	.163029+01	.162935+01	.157146+01	.155924+01	.154507+01
58	.159000+03	.166619+01	.163120+01	.164255+01	.164419+01	.164909+01
59	.154000+03	.180312+01	.170117+01	.172349+01	.173044+01	.173920+01
60	.150000+03	.183011+01	.190506+01	.184610+01	.182246+01	.182221+01
61	.142000+03	.189538+01	.175516+01	.182698+01	.184750+01	.187947+01
62	.134000+03	.186306+01	.203561+01	.193135+01	.188777+01	.190305+01
63	.122000+03	.176946+01	.169050+01	.177434+01	.178822+01	.183029+01
64	.109000+03	.148404+01	.164842+01	.170064+01	.164033+01	.162052+01
65	.949000+02	.119156+01	.111966+01	.126509+01	.131772+01	.133342+01
66	.849000+02	.981093+00	.126346+01	.114537+01	.110994+01	.108476+01
67	.758000+02	.759434+00	.698729+00	.813085+00	.851320+00	.862485+00
68	.682000+02	.551062+00	.820139+00	.710212+00	.679618+00	.651227+00
69	.635000+02	.438280+00	.231986+00	.423776+00	.473783+00	.486588+00
70	.598000+02	.279296+00	.594575+00	.437380+00	.388945+00	.345623+00
71	.581000+02	.237834+00	-.360034-01	.160372+00	.234539+00	.253376+00
72	.587000+02	.235595+00	.511672+00	.328367+00	.271787+00	.237143+00
73	.595000+02	.275524+00	-.404817-01	.156879+00	.221258+00	.250751+00
74	.599000+02	.247940+00	.591530+00	.371665+00	.302165+00	.261446+00
75	.612000+02	.356063+00	-.956501-01	.169453+00	.257803+00	.300740+00
76	.622000+02	.328169+00	.807785+00	.497341+00	.393691+00	.340124+00
77	.620000+02	.433273+00	-.151446+00	.203332+00	.324941+00	.387975+00
78	.637000+02	.465672+00	.101799+01	.638979+00	.505755+00	.445721+00
79	.609000+02	.446827+00	-.866470-01	.275284+00	.399367+00	.461582+00
80	.617000+02	.541131+00	.980301+00	.656086+00	.545470+00	.499597+00
81	.590000+02	.476486+00	.101962+00	.373193+00	.459122+00	.505120+00
82	.569000+02	.490132+00	.851010+00	.605876+00	.525614+00	.479348+00
83	.550000+02	.451570+00	.129254+00	.356985+00	.436188+00	.474625+00
84	.535000+02	.524485+00	.773885+00	.583313+00	.518267+00	.493457+00
85	.512000+02	.458518+00	.275084+00	.419363+00	.462768+00	.489053+00
86	.475000+02	.447053+00	.641952+00	.515841+00	.475609+00	.454204+00
87	.455000+02	.413575+00	.252153+00	.370927+00	.410215+00	.433371+00
88	.423000+02	.396640+00	.574997+00	.461738+00	.421547+00	.404497+00
89	.394000+02	.319288+00	.218283+00	.311404+00	.342740+00	.357776+00
90	.366000+02	.307685+00	.420293+00	.349089+00	.326109+00	.315393+00
91	.346000+02	.243915+00	.195077+00	.248892+00	.265388+00	.273774+00
92	.320000+02	.206338+00	.292753+00	.247669+00	.233550+00	.221680+00
93	.308000+02	.158938+00	.119924+00	.161733+00	.178021+00	.181572+00
94	.298000+02	.163684+00	.197952+00	.173525+00	.166597+00	.161959+00
95	.295000+02	.138160+00	.129416+00	.143754+00	.147630+00	.148271+00
96	.286000+02	.126659+00	.146904+00	.137241+00	.135565+00	.131135+00

97	.286000+02	.128947+00	.106414+00	.120674+00	.126791+00	.129533+00
98	.288000+02	.142012+00	.151490+00	.140813+00	.136082+00	.135279+00
99	.286000+02	.123136+00	.132544+00	.132564+00	.132417+00	.130777+00
100	.235000+02	.133440+00	.113727+00	.123434+00	.127932+00	.127739+00
101	.289000+02	.137271+00	.153154+00	.141239+00	.137673+00	.135051+00
102	.290000+02	.153683+00	.121389+00	.137443+00	.143121+00	.145601+00
103	.293000+02	.163282+00	.135977+00	.167647+00	.161362+00	.157793+00
104	.293000+02	.176538+00	.140588+00	.160136+00	.166738+00	.169258+00
105	.293000+02	.188575+00	.212468+00	.192534+00	.186941+00	.184700+00
106	.294000+02	.220392+00	.164661+00	.191376+00	.198591+00	.203880+00
107	.287000+02	.202518+00	.277122+00	.233511+00	.219097+00	.210248+00
108	.281000+02	.237065+00	.127914+00	.189166+00	.210914+00	.221968+00
109	.278000+02	.242651+00	.346215+00	.275310+00	.250024+00	.241802+00
110	.261000+02	.247481+00	.139086+00	.209739+00	.232567+00	.248923+00
111	.249000+02	.240043+00	.355975+00	.281135+00	.253574+00	.244534+00
112	.224000+02	.197238+00	.124221+00	.187169+00	.207392+00	.217560+00
113	.207000+02	.176144+00	.270255+00	.214546+00	.196812+00	.186424+00
114	.192000+02	.148384+00	.820339-01	.135521+00	.153449+00	.161655+00
115	.179000+02	.129084+00	.214735+00	.164068+00	.147731+00	.138613+00
116	.179000+02	.111267+00	.434341-01	.945953-01	.111685+00	.119200+00
117	.162000+02	.926072-01	.179101+00	.127658+00	.111084+00	.100585+00
118	.156000+02	.850984-01	.611356-02	.612730-01	.806624-01	.878161-01
119	.158000+02	.844097-01	.164083+00	.111224+00	.940922-01	.830009-01
120	.158000+02	.857698-01	.489619-02	.583649-01	.772763-01	.842882-01
121	.162000+02	.104273+00	.166641+00	.118894+00	.103765+00	.961277-01
122	.164000+02	.119697+00	.419048-01	.886250-01	.103699+00	.112752+00
123	.164000+02	.129221+00	.197489+00	.148802+00	.131485+00	.122800+00
124	.159000+02	.125052+00	.609521-01	.105075+00	.121152+00	.128343+00
125	.160000+02	.159223+00	.169151+00	.157809+00	.146820+00	.143709+00
126	.154000+02	.145321+00	.129295+00	.144613+00	.148344+00	.151653+00
127	.144000+02	.154748+00	.161346+00	.153805+00	.151714+00	.150595+00
128	.139000+02	.148464+00	.148150+00	.150454+00	.151080+00	.153048+00
129	.126000+02	.155336+00	.148779+00	.150860+00	.151849+00	.152258+00
130	.122000+02	.154366+00	.161893+00	.157198+00	.155001+00	.142889+00
131	.109000+02	.149861+00	.146939+00	.150355+00	.152900+00	.131625+00
132	.103000+02	.173593+00	.152883+00	.158779+00	.161875+00	.191206+00
133	.976000+01	.189435+00	.194302+00	.185776+00	.150314+00	.370920+00
134	.951000+01	.184567+00	.184567+00	.000000	.000000	.562712+00

BACK CHECKS

INPUT RECOMPUTED

I	XI(I)	ZI(I)			
7	.60760+04	.60833+04	57	.16000+03	.16285+03
8	.57770+04	.57555+04	58	.15900+03	.15931+03
9	.55770+04	.55991+04	59	.15400+03	.15624+03
10	.53610+04	.53151+04	60	.15000+03	.15022+03
11	.52220+04	.52649+04	61	.14200+03	.14255+03
12	.51090+04	.50973+04	62	.13400+03	.13409+03
13	.49300+04	.49539+04	63	.12200+03	.12225+03
14	.48260+04	.48071+04	64	.10900+03	.10985+03
15	.46980+04	.47003+04	65	.94900+02	.95175+02
16	.45940+04	.45739+04	66	.84900+02	.85983+02
17	.44900+04	.44881+04	67	.75800+02	.75419+02
18	.43840+04	.43733+04	68	.68200+02	.69085+02
19	.42880+04	.42983+04	69	.63500+02	.63201+02
20	.41680+04	.41793+04	70	.59800+02	.60219+02
21	.41480+04	.41641+04	71	.58100+02	.58481+02
22	.41140+04	.41064+04	72	.58700+02	.59777+02
23	.40530+04	.40657+04	73	.59500+02	.59283+02
24	.39790+04	.39807+04	74	.59900+02	.61252+02
25	.38070+04	.38268+04	75	.61200+02	.61092+02
26	.34680+04	.34734+04	76	.62200+02	.63436+02
27	.29470+04	.29710+04	77	.62000+02	.61060+02
28	.22210+04	.22003+04	78	.63700+02	.65499+02
29	.13670+04	.13293+04	79	.60900+02	.60169+02
30	.79100+03	.76682+03	80	.61700+02	.64256+02
31	.58300+03	.57932+03	81	.59000+02	.58137+02
32	.55800+03	.55513+03	82	.56900+02	.58563+02
33	.57700+03	.57777+03	83	.55000+02	.53951+02
34	.60900+03	.60873+03	84	.53500+02	.55344+02
35	.62800+03	.63953+03	85	.51200+02	.50700+02
36	.63900+03	.64291+03	86	.47500+02	.49136+02
37	.63000+03	.63142+03	87	.45500+02	.45100+02
38	.61400+03	.61612+03	88	.42300+02	.43829+02
39	.58900+03	.58949+03	89	.39400+02	.39237+02
40	.55100+03	.55315+03	90	.36600+02	.38287+02
41	.50000+03	.50053+03	91	.34600+02	.34447+02
42	.43300+03	.43239+03	92	.32000+02	.33249+02
43	.36800+03	.36940+03	93	.30800+02	.30847+02
44	.30500+03	.30294+03	94	.29800+02	.31112+02
45	.25200+03	.25226+03	95	.29500+02	.29707+02
46	.20700+03	.20566+03	96	.28600+02	.29653+02
47	.17600+03	.17619+03	97	.28600+02	.29247+02
48	.15800+03	.15678+03	98	.28800+02	.29821+02
49	.15000+03	.15142+03	99	.28600+02	.29027+02
50	.14800+03	.14752+03	100	.28500+02	.29463+02
51	.15200+03	.15435+03	101	.28900+02	.29529+02
52	.15500+03	.15405+03	102	.29000+02	.29795+02
53	.15800+03	.15932+03	103	.29300+02	.29902+02
54	.16000+03	.15877+03	104	.29300+02	.30029+02
55	.16500+03	.16685+03	105	.29300+02	.30221+02
56	.16300+03	.16225+03	106	.29400+02	.29983+02

107	.28700+02	.29450+02
108	.28100+02	.29189+02
109	.27300+02	.28519+02
110	.26100+02	.26693+02
111	.24900+02	.26281+02
112	.22400+02	.22509+02
113	.20700+02	.22001+02
114	.19200+02	.19557+02
115	.17900+02	.19204+02
116	.17000+02	.17167+02
117	.16200+02	.17371+02
118	.15800+02	.16407+02
119	.15800+02	.16970+02
120	.15800+02	.16623+02
121	.16200+02	.17601+02
122	.16400+02	.17245+02
123	.16400+02	.17726+02
124	.15900+02	.16658+02
125	.16000+02	.17817+02
126	.15400+02	.16269+02
127	.14400+02	.16060+02
128	.13900+02	.15013+02
129	.12800+02	.14465+02
130	.12200+02	.13669+02
131	.10900+02	.12612+02
132	.10300+02	.11313+02
133	.97600+01	.10611+02
134	.95100+01	.76184+01

END OF RUN

@FIN NIF@

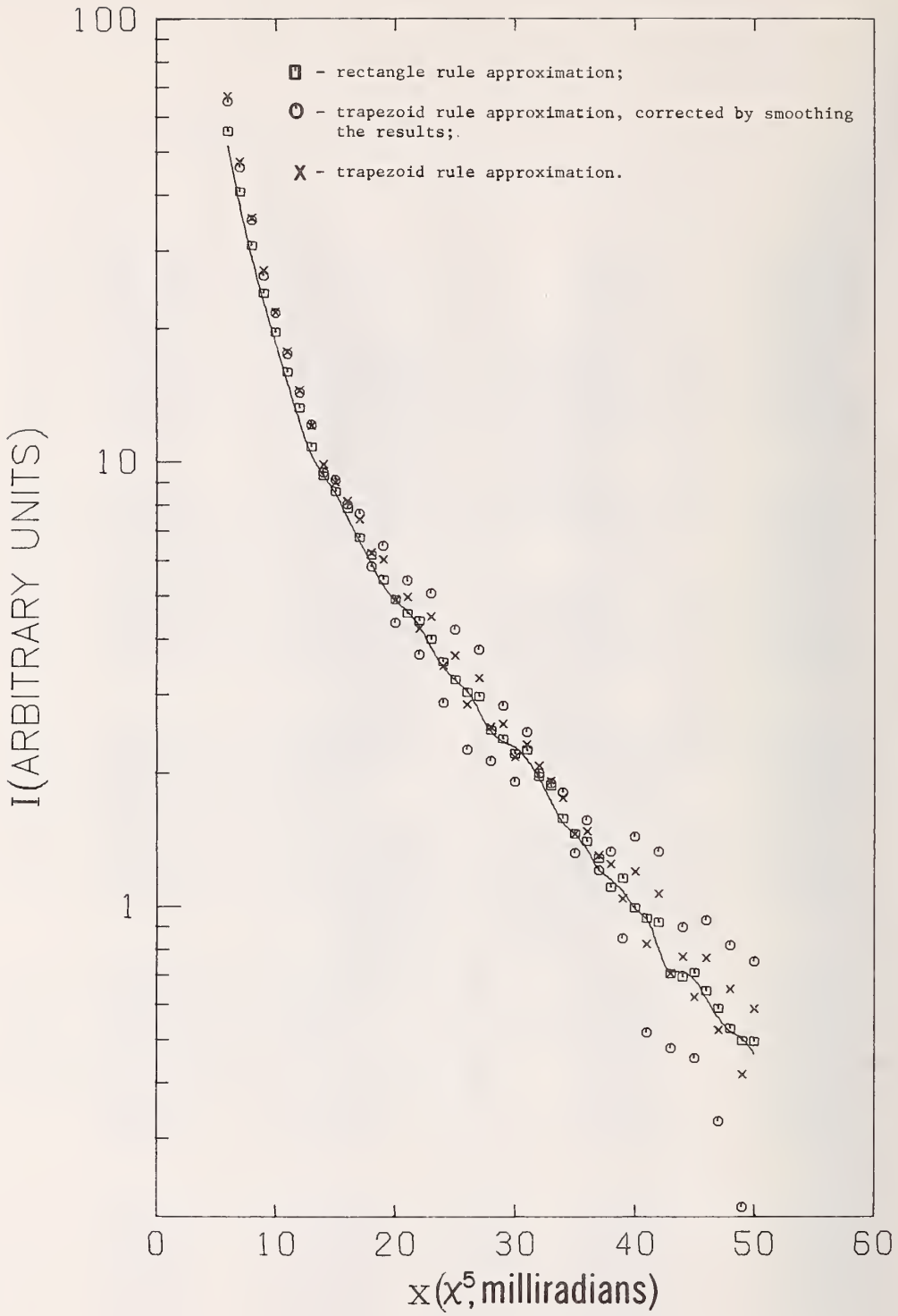


Figure 1. Scattering from melt-crystallized polyethylene, at very low angles ($2\theta \leq 0.008$ radians). Weighting function is constant. Solid line-Nÿstrom (forward) interpolation.

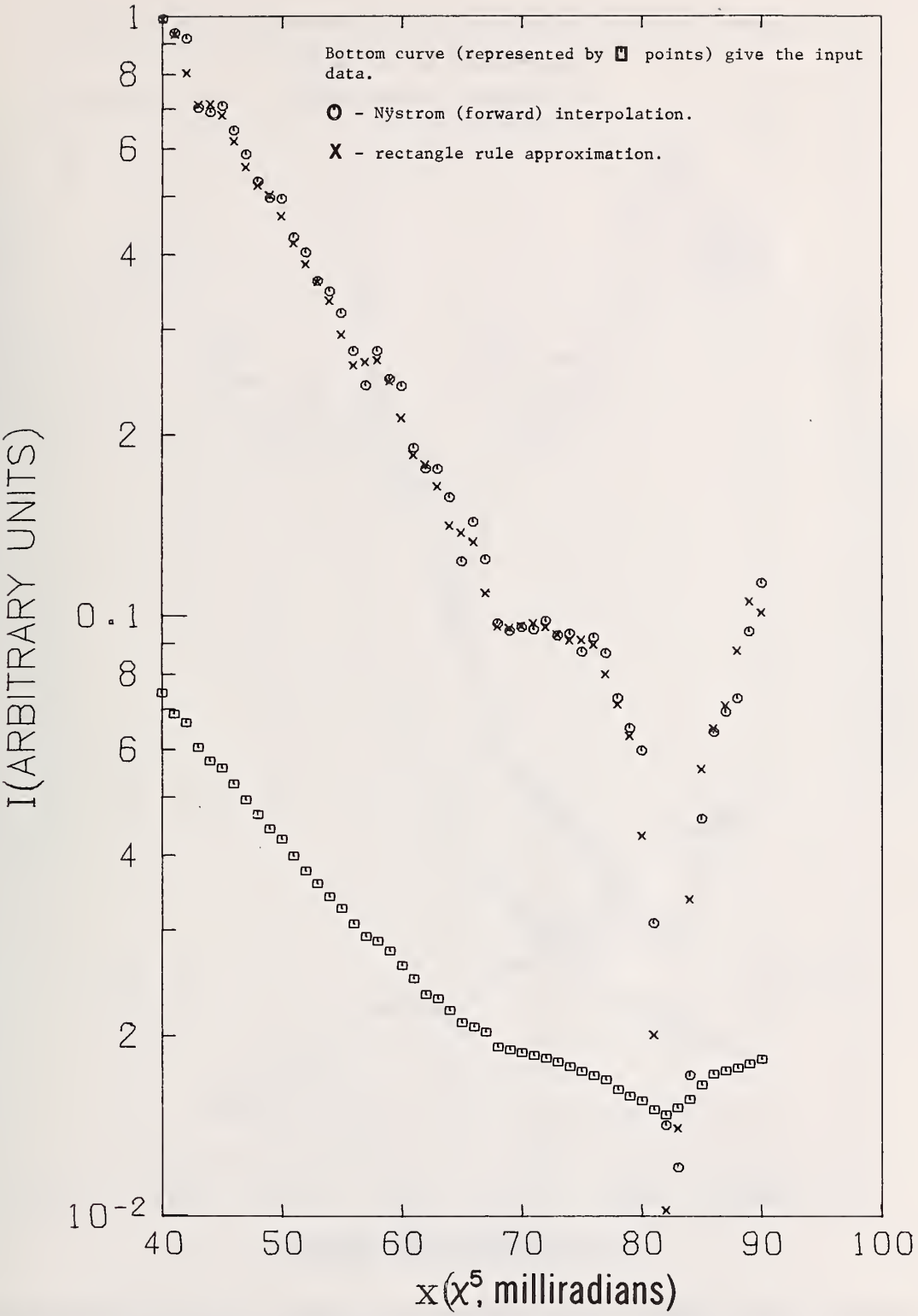


Figure 2. Data from the same experiment as in Figure 1, but for scattering angles in the range of ($0.008 \lesssim 2\theta \lesssim .02$ radians).

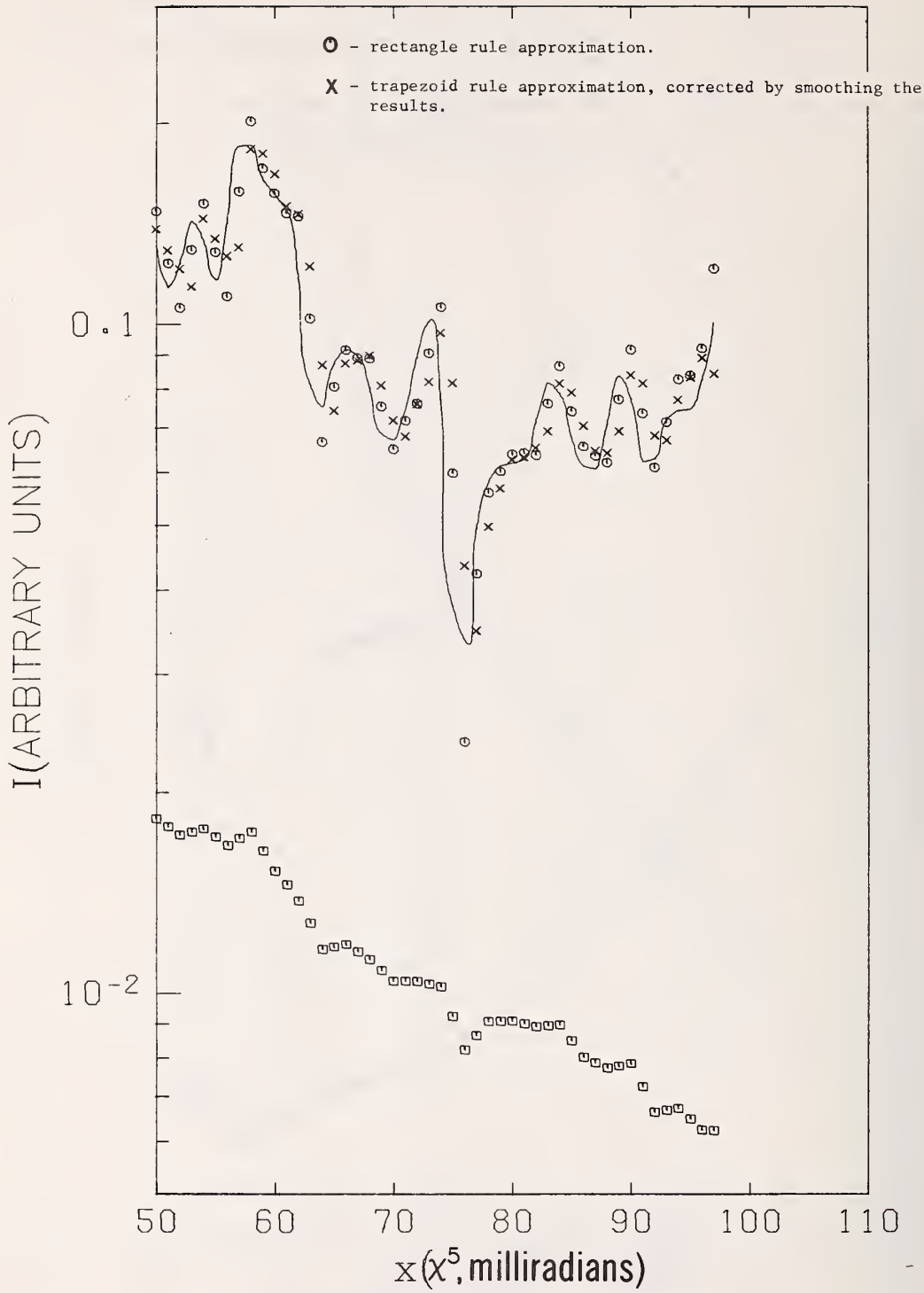


Figure 3. Unsmearing of experimental data which were not smoothed out for random noise effects. Bottom curve (represented by ▯ points) show the unsmoothed input data. Solid line-Nÿstrom (forward) interpolation.

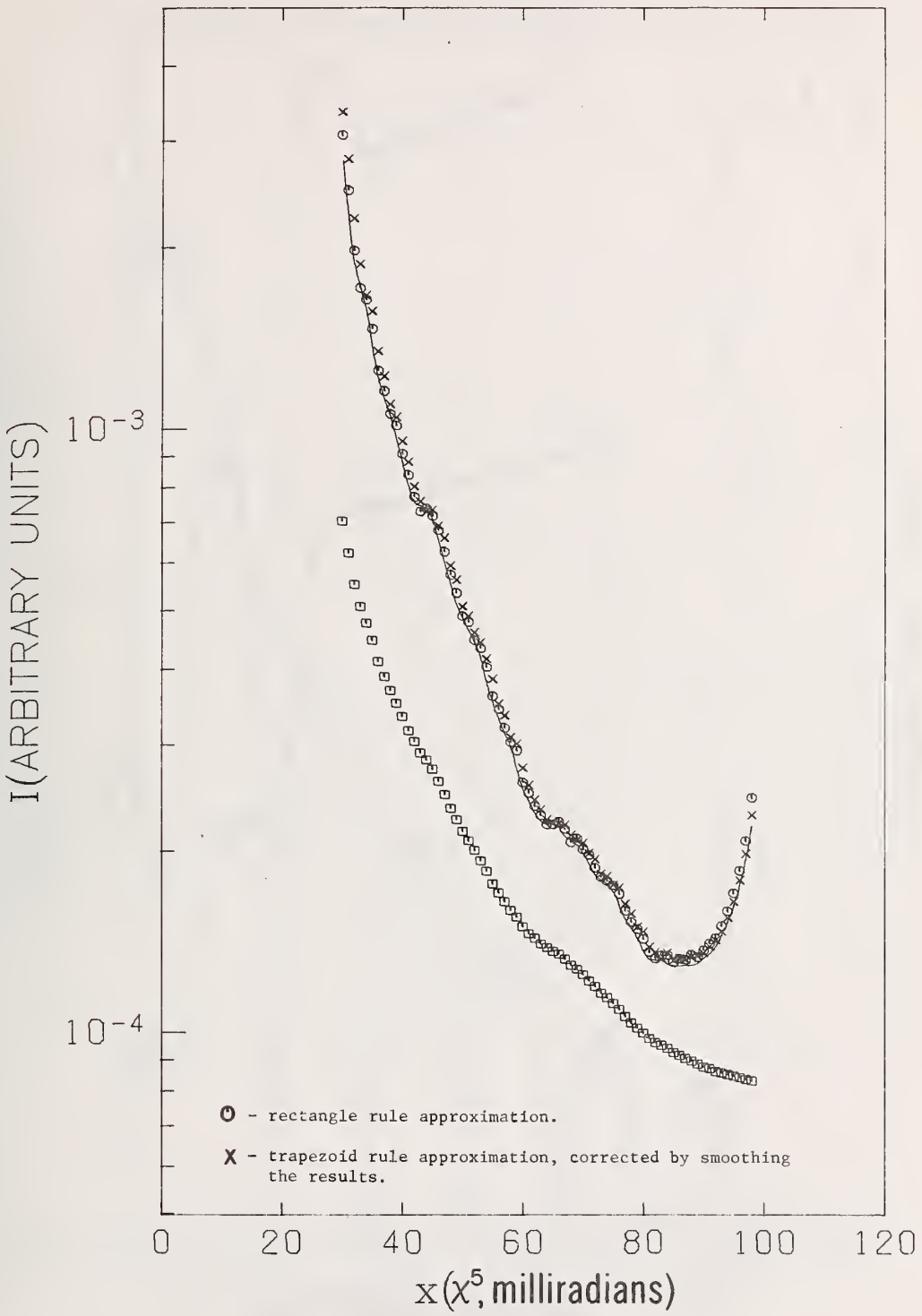


Figure 4. Scattering from melt-crystallized polyethylene. N=100, trapezoid weighting function. Bottom curve (represented by points) show the input data. Solid line-Nÿstrom (forward) interpolation.

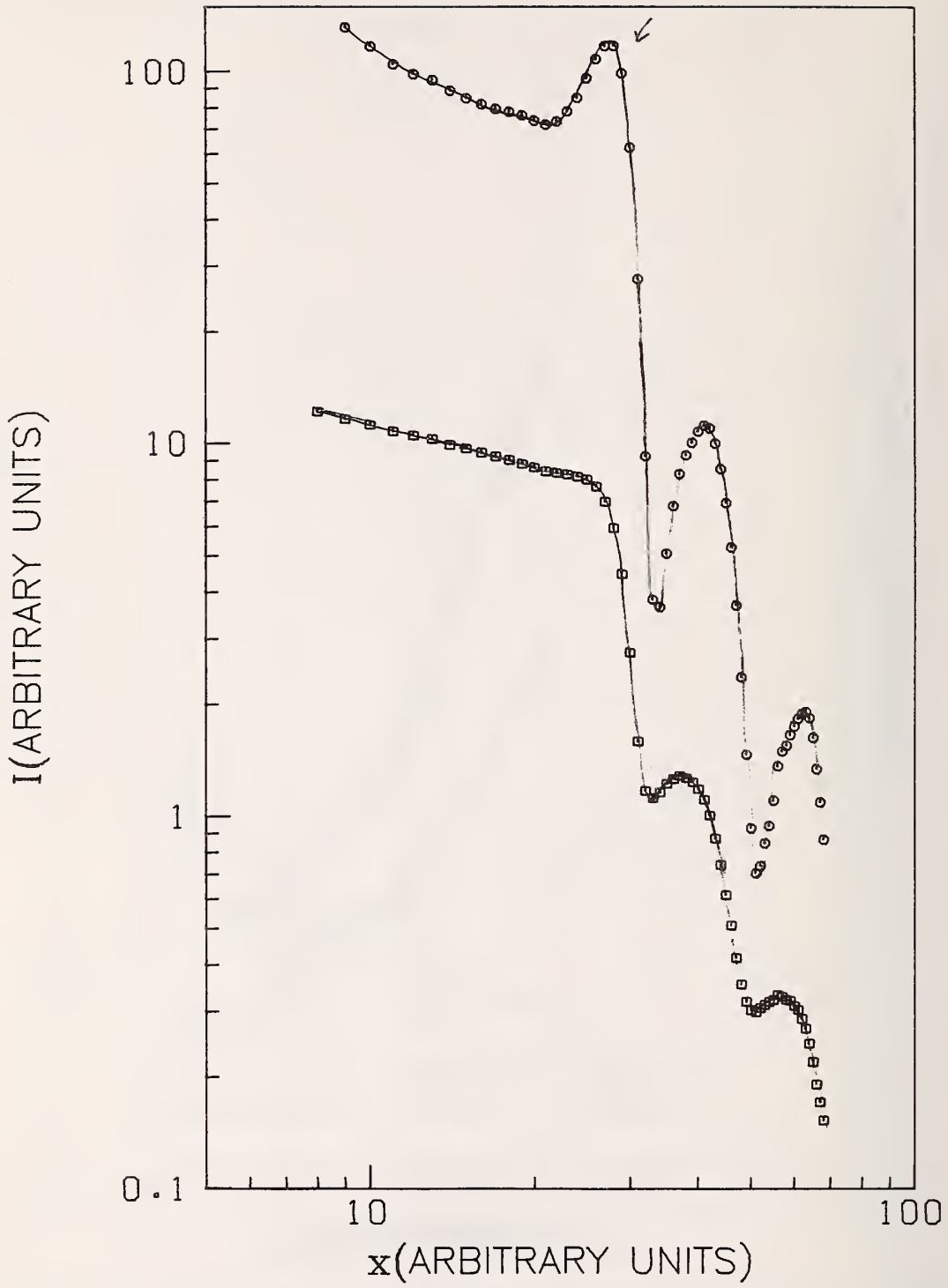


Figure 5. Scattering from triblock copolymer. Constant weighting function. Bottom curve shows the input data. Upper curve shows desmeared data, Nÿstrom (forward) interpolation.

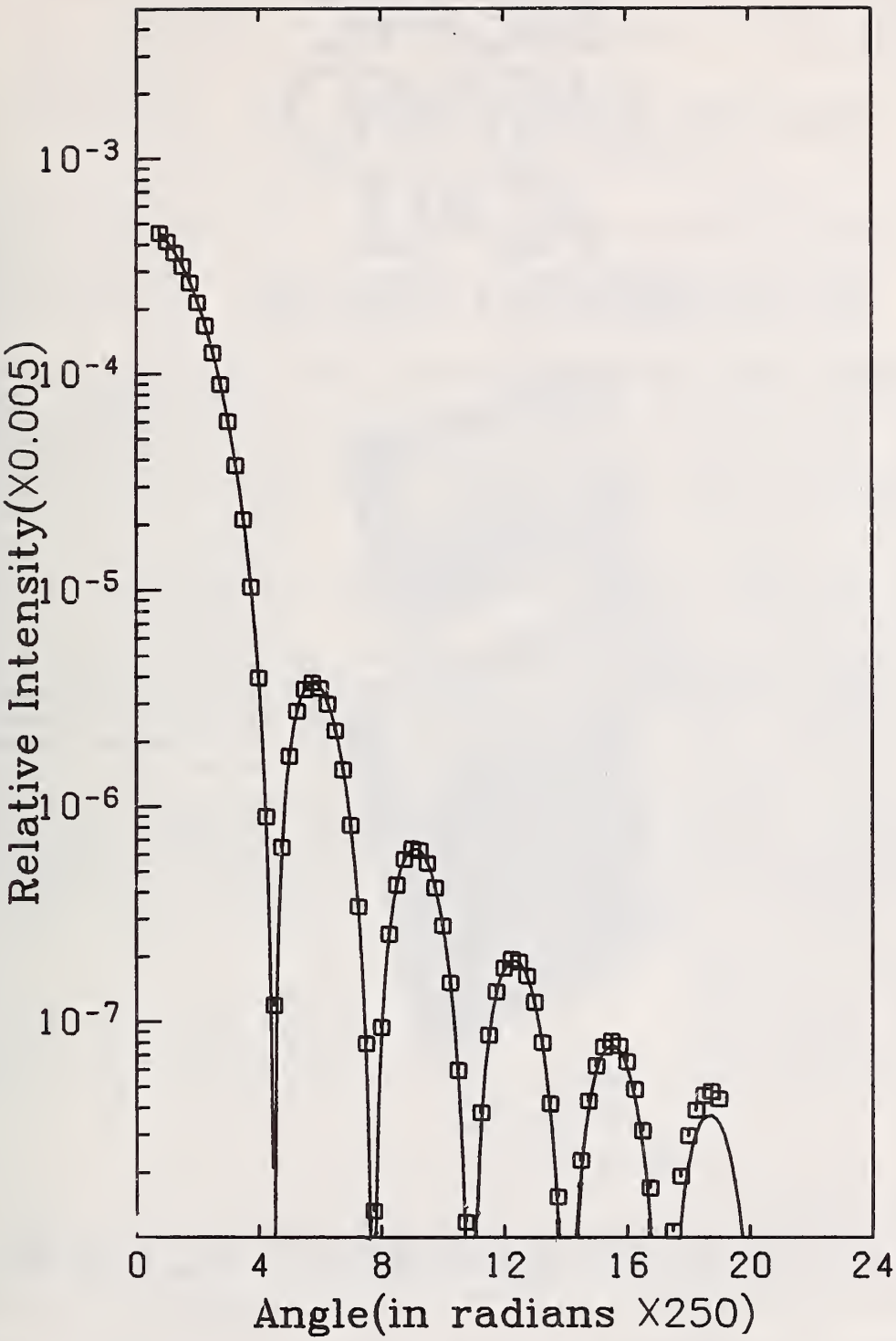


Figure 6. Scattering intensities by spheres with uniform electron densities. Solid line-exact results. Points represent computed results, based on Nÿstrom (forward) interpolation.

☆U.S. GOVERNMENT PRINTING OFFICE: 1977-240-848/52

NBS-114A (REV. 7-73)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET	1. PUBLICATION OR REPORT NO. NBS TN-936	2. Gov't Accession No.	3. Recipient's Accession No.

4. TITLE AND SUBTITLE

A FORTRAN PROGRAM FOR DESMEARING SMALL-ANGLE X-RAY SCATTERING CURVES.

5. Publication Date: April 1977

6. Performing Organization Code

7. AUTHOR(S): Jacob Mazur

8. Performing Organ. Report No.

9. PERFORMING ORGANIZATION NAME AND ADDRESS

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

10. Project/Task/Work Unit No. 3110110

11. Contract/Grant No.

12. Sponsoring Organization Name and Complete Address *(Street, City, State, ZIP)*

Same as 9.

13. Type of Report & Period Covered: Final

14. Sponsoring Agency Code

15. SUPPLEMENTARY NOTES

16. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

A FORTRAN program is presented that corrects for collimation effects in small angle x-ray scattering measurements. The method employed in calculations is based on numerical solution of an integral equation, which is written as a Volterra equation of the first kind. This equation is reduced to a system of simultaneous equations, which are solved by a matrix inversion method. Several different algorithms are tried, based on different interpolating polynomials for I(x), the true scattering intensity which is expressed as a function of the scattering angle x. The problem of numerical stability, which is inherent to the computations algorithms employed in the algebraic solution of integral equation is discussed and exemplified with actual computations.

17. KEY WORDS *(six to twelve entries; alphabetical order; capitalize only the first letter of the first key word unless a proper name; separated by semicolons)*

FORTRAN program; integral equation, interpolating polynomial; numerical stability; slit correction; small angle scattering; unsmearing.

18. AVAILABILITY

[X] Unlimited

[ ] For Official Distribution. Do Not Release to NTIS

[X] Order From Sup. of Doc., U.S. Government Printing Office Washington, D.C. 20402, SD Cat. No. C13.46:936

[ ] Order From National Technical Information Service (NTIS) Springfield, Virginia 22151

19. SECURITY CLASS (THIS REPORT): UNCLASSIFIED

21. NO. OF PAGES: 83

20. SECURITY CLASS (THIS PAGE): UNCLASSIFIED

22. Price $1.45

USCOMM-DC 29042-P74

PATENT REVIEW IS NOT REQUIRED.

SINGLE CRYSTAL DATA

REVISED! UPDATED!

In 1954, the first edition of CRYSTAL DATA (Determinative Tables and Systematic Tables) was published as Memoir 60 of the Geological Society of America. In 1960, the second edition of the Determinative Tables was issued as Monograph 5 of the American Crystallographic Association, and in 1967, the Systematic Tables were issued as Monograph 6. These editions proved extremely valuable to crystallographers throughout the world. Recognizing the need for updated crystallographic information, the National Bureau of Standards Office of Standard Reference Data has sponsored the issuance of a new edition.

This, the THIRD EDITION, should be of particular interest not only to crystallographers but also to chemists, mineralogists, physicists and individuals in related fields of study. The current edition, which comprises two volumes, Organic and Inorganic, is a thoroughly revised and updated work, containing over 25,000 entries.

The entries are listed, *within each crystal system,* according to increasing values of a determinative number: a/*b* ratio in trimetric systems, c/a ratio in dimetric systems, and cubic cell edge a, in the isometric system. In addition, the following information is given:

INORGANIC VOLUME $50.00

ORGANIC VOLUME $30.00

Plus shipping and handling

axial ratio(s) and interaxial angles not fixed by symmetry, cell dimensions, space group or diffraction aspect, number of formula units per unit cell, crystal structure, (whether determined), measured density and x-ray calculated density. Also listed is the name of the compound and synonym(s), chemical formula, literature reference and transformation matrix. When available, the crystal structure type, crystal habit, cleavages, twinning, color, optical properties, indices of refraction, optical orientation, melting point and transition point are also listed.

THIS EDITION culminates years of effort by J. D. H. Donnay, Johns Hopkins University, Helen M. Ondik, National Bureau of Standards, Sten Samson, California Institute of Technology, Quintin Johnson, Lawrence Radiation Laboratory, Melvin H. Mueller, Argonne National Laboratory, Gerard M. Wolten, Aerospace Corporation, Mary E. Mrose, U.S. Geological Survey, Olga Kennard and David G. Watson, Cambridge University, England and Murray Vernon King, Massachusetts General Hospital.

Shipments are made via insured parcel post. Additional charges for shipments by air or commercial carrier.
TERMS: Domestic—30 days Foreign—prepayment required. Address all orders to:
JOINT COMMITTEE ON POWDER DIFFRACTION STANDARDS 1601 Park Lane, Swarthmore, Pennsylvania 19081

Please accept my order for CRYSTAL DATA, DETERMINATIVE TABLES, Third Edition, Donnay/Ondik.

☐ **Organic Volume**

☐ **Inorganic Volume**

Ship to: _____

_____

Signature _____

NATIONAL STANDARD NSRDS REFERENCE DATA SYSTEM

JCPDS

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH reports National Bureau of Standards research and development in physics, mathematics, and chemistry. It is published in two sections, available separately:

- **Physics and Chemistry (Section A)**

Papers of interest primarily to scientists working in these fields. This section covers a broad range of physical and chemical research, with major emphasis on standards of physical measurement, fundamental constants, and properties of matter. Issued six times a year. Annual subscription: Domestic, $17.00; Foreign, $21.25.

- **Mathematical Sciences (Section B)**

Studies and compilations designed mainly for the mathematician and theoretical physicist. Topics in mathematical statistics, theory of experiment design, numerical analysis, theoretical physics and chemistry, logical design and programming of computers and computer systems. Short numerical tables. Issued quarterly. Annual subscription: Domestic, $9.00; Foreign, $11.25.

DIMENSIONS/NBS (formerly Technical News Bulletin)—This monthly magazine is published to inform scientists, engineers, businessmen, industry, teachers, students, and consumers of the latest advances in science and technology, with primary emphasis on the work at NBS. The magazine highlights and reviews such issues as energy research, fire protection, building technology, metric conversion, pollution abatement, health and safety, and consumer product performance. In addition, it reports the results of Bureau programs in measurement standards and techniques, properties of matter and materials, engineering standards and services, instrumentation, and automatic data processing.

Annual subscription: Domestic, $12.50; Foreign, $15.65.

NONPERIODICALS

Monographs—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a world-wide program coordinated by NBS. Program under authority of National Standard Data Act (Public Law 90-396).

NOTE: At present the principal publication outlet for these data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St. N.W., Wash. D. C. 20056.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The purpose of the standards is to establish nationally recognized requirements for products, and to provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, D.C. 20402.*

Order **following** *NBS publications—NBSIR's and FIPS from the National Technical Information Services, Springfield, Va. 22161.*

Federal Information Processing Standards Publications (FIPS PUBS)—Publications in this series collectively constitute the Federal Information Processing Standards Register. Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services (Springfield, Va. 22161) in paper copy or microfiche form.

BIBLIOGRAPHIC SUBSCRIPTION SERVICES

The following current-awareness and literature-survey bibliographies are issued periodically by the Bureau:

Cryogenic Data Center Current Awareness Service. A literature survey issued biweekly. Annual subscription: Domestic, $20.00; Foreign, $25.00.

Liquified Natural Gas. A literature survey issued quarterly. Annual subscription: $20.00.

Superconducting Devices and Materials. A literature survey issued quarterly. Annual subscription: $20.00. Send subscription orders and remittances for the preceding bibliographic services to National Bureau of Standards, Cryogenic Data Center (275.02) Boulder, Colorado 80302.

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Washington, D.C. 20234

OFFICIAL BUSINESS

Penalty for Private Use, $300

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE
COM-215

SPECIAL FOURTH-CLASS RATE
BOOK